

(51) Int.Cl. ⁶	識別記号	F I		
G 0 6 F 17/30		G 0 6 F 15/40	3 8 0 E	
12/00	5 1 3	12/00	5 1 3 J	
	5 4 7		5 4 7 Q	
		15/40	3 8 0 D	
		15/403	3 4 0 D	
審査請求 未請求 予備審査請求 有 (全 79 頁)				

(21) 出願番号 特願平8-532654
 (86) (22) 出願日 平成8年(1996) 4月23日
 (85) 翻訳文提出日 平成9年(1997) 10月24日
 (86) 国際出願番号 P C T / U S 9 6 / 0 5 6 7 8
 (87) 国際公開番号 W O 9 6 / 3 4 3 5 0
 (87) 国際公開日 平成8年(1996) 10月31日
 (31) 優先権主張番号 0 8 / 4 2 8 , 0 0 3
 (32) 優先日 1995年4月24日
 (33) 優先権主張国 米国 (U S)
 (31) 優先権主張番号 0 8 / 5 2 1 , 6 6 7
 (32) 優先日 1995年8月31日
 (33) 優先権主張国 米国 (U S)

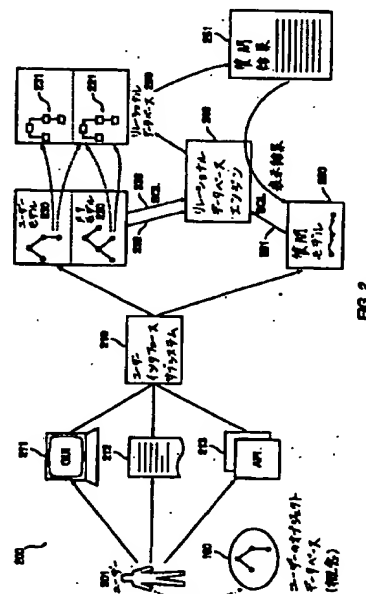
(71) 出願人 アスペクト・ディベロップメント・インコーポレイテッド
 アメリカ合衆国94043カリフォルニア州マウンテン・ビュー、チャールストン・ロード1300番
 (72) 発明者 アルトフ、ジェイムズ
 アメリカ合衆国94022カリフォルニア州ロス・アルトス、ロス・ニノス・ウェイ709番
 (72) 発明者 リー、ソン
 アメリカ合衆国95138カリフォルニア州サン・ノゼ、ローズウェル・ウェイ95番
 (74) 代理人 弁理士 青山 稔 (外1名)

最終頁に続く

(54) 【発明の名称】 データベース構造に適したオブジェクトのモデリング、リレーショナルデータベース構造への翻訳、それらへの流動的なサーチ

(57) 【要約】

データベース構造に適したオブジェクトのモデリング、リレーショナルデータベース構造への翻訳、およびそれらへの流動的なサーチのための方法およびシステム。ユーザーのオブジェクトデータベースを、創造し、エディットし、そして扱う(これらのリレーショナルデータベース構造のためのデータに流動的に翻訳される)ためと、前記オブジェクトデータに適用される質問を創造し、エディットし、扱う(それらのリレーショナルデータベースに適用されるべき質問に流動的に翻訳される)ために、ユーザーは、ユーザーのオブジェクトデータベース(1組みのリレーショナルデータベース構造への流動的に翻訳される)を創造し、エディットし、処理してもよい。ユーザーのオブジェクトデータベースのメタモデル、それ自身はひとつのオブジェクトデータベースであり、又、操作のために、リレーショナルデータベースエンジンにより、それ自身が1組みのリレーショナルデータベース構造に翻訳される。メタモデルは、1組みのクラス、オブジェクトとクラス間の関係を含み、クラスと、システムのクラス間の関係をモデリングする。これ



【特許請求の範囲】

(1) ユーザーのオブジェクトデータベースの記述を受け取るための手段であって、前記ユーザーのオブジェクトデータベースは1組みのクラスと、対の前記クラス間の1組みの関係を持っている、手段と、

前記記述に応答して前記ユーザーのオブジェクトデータベースのモデルを創造するための手段と、

前記モデルに응答してリレーショナルデータベースを創造するための手段であって、前記リレーショナルデータベースは1組みのテーブル、前記テーブルのためのキーおよび、前記クラス、オブジェクトおよび前記ユーザーのオブジェクトデータベースの関係を実行する組みのテーブル間の関係を持っている、手段と、

前記ユーザーのオブジェクトデータベースに対する1組みのデータベースを受け取るための手段と、

前記ユーザーのオブジェクトデータベースに対する前記組みのデータオブジェクトを、前記リレーショナルデータベースに対する1組みの記録に翻訳し、そして、前記記録の組みを前記リレーショナルデータベースに挿入するための手段と、

ユーザーのオブジェクトデータベースの前記記述にアップデートされたものを受け取るための手段と、

응答アップデートされたものに응答して前記モデルをアップデートするための手段と、

前記モデルをアップデートするための前記手段に응答して前記リレーショナルデータベースをアップデートするための手段と、および

前記リレーショナルデータベースをアップデートするための前記手段に응答して前記リレーショナルデータベースに対する前記組の記録をアップデートするための手段と、

を含むシステム。

(2) 1組みのリレーショナルデータベースコマンドを受け取るための手段を含むリレーショナルデータベースサーバーを含み、

翻訳し、挿入するための前記手段は1組みのリレーショナルデータベースコマ

ンドを発生させるための手段を含む請求の範囲第1項記載のシステム。

(3) 挿入するための前記手段は、前記テーブルに対する記録を発生させるための手段と、

前記記録に対する前記キーのための値を発生させるための手段と、

組みのテーブル間の前記関係に対応する前記記録のための前記キーに対する前記値を生じさせる手段とを含む請求の範囲第1項記載のシステム。

(4) 挿入するための前記手段は、前記組みの記録を前記リレーショナルデータベースに挿入している間、前記テーブル、前記テーブルに対するキーおよび組みのテーブル間の前記関係を維持するための手段を含む請求の範囲第1項に記載のシステム。

(5) ユーザーのオブジェクトデータベースの記述を受け取るための手段であって、前記ユーザーのオブジェクトデータベースは1組みのクラスと、対の前記クラス間の1組みの関係を持っている、手段と、

前記記述に応答して前記ユーザーのオブジェクトデータベースのモデルを創造するための手段であって、前記モデルは、一つのオブジェクトデータベースを含み、このオブジェクトデータベースは、

第1のクラスのオブジェクト、

前記ユーザーのオブジェクトデータベース内の前記各々の一つのクラスに対応する前記第1のクラスのオブジェクト、

第2のクラスのオブジェクト、そして、

前記ユーザーのオブジェクトデータベース内の前記各々の一つの関係に対応する前記第2のクラスのオブジェクト、を含む手段と、そして、

前記モデルに応答してリレーショナルデータベースを創造するための手段であって、前記リレーショナルデータベースは1組みのテーブル、前記テーブルのためのキーおよび、前記クラス、オブジェクトおよび前記ユーザーのオブジェクトデータベースの関係を実行する組みのテーブル間の関係を持っている、手段と、を含むシステム。

(6) 1組みのリレーショナルデータベースコマンドを受け取るための手段を含

むりレシヨナルデータベースサーバーを含み、

翻訳し、挿入するための前記手段は1組みのリレシヨナルデータベースコマンドを発生させるための手段を含む請求の範囲第5項記載のシステム。

(7) 前記サーバーは、前記コマンドの第2のソースからの1組みのリレシヨナルデータベースコマンドを受け取るための手段を含む請求の範囲第5項記載のシステム。

(8) ユーザーのオブジェクトデータベースの記述を受け取るための手段であって、前記ユーザーのオブジェクトデータベースは1組みのクラスと、対の前記クラス間の1組みの関係とを持ち、前記記述へアップデートされたものを受け取るための手段と、

前記記述に応答して前記ユーザーのオブジェクトデータベースのモデルを創造し、かつ、前記アップデートされたものに応答して前記モデルをアップデートするための手段と、

前記モデル、1組みのテーブルを持つ前記リレシヨナルデータベース、前記テーブルのためのキー、および、前記クラスを実行する前記テーブル間の関係、オブジェクトおよびユーザーのデータベースオブジェクトの関係とに回答するリレシヨナルデータベースを創造するためと、前記モデルをアップデートするための前記手段に回答して前記リレシヨナルデータベースをアップデートするための手段とを備えるシステム。

(9) トリガー用信号を受け取るための手段を含み、前記リレシヨナルデータベースをアップデートするための前記手段は、前記トリガー用信号に回答して機能する請求の範囲第8項記載のシステム。

(10) ユーザーのオブジェクトデータベースの記述を受け取るための手段であって、前記ユーザーのオブジェクトデータベースは1組みのクラスと、対の前記クラス間の1組みの関係を持っている、手段と、

前記記述に回答して前記ユーザーのオブジェクトデータベースのモデルを創造するための手段と、

前記モデルに回答してリレシヨナルデータベースを創造するための手段であ

っ

て、前記リレーショナルデータベースは1組みのテーブル、前記テーブルのためのキーおよび、前記クラス、オブジェクトおよび前記ユーザーのオブジェクトデータベースの関係を実行する組みのテーブル間の関係を持っている、手段と、

前記ユーザーのオブジェクトデータベースに対する質問の記述を受け取るための手段と、

前記質問を、前記リレーショナルデータベースへのアプリケーションに対して適したリレーショナルデータベースの質問に翻訳するための手段と、

を備えるシステム。

(11) 前記リレーショナルデータベースへ前記リレーショナルデータベースの質問を与えるための手段と、与えるための前記手段の出力を示すための手段とを含む請求の範囲第10項記載のシステム。

(12) データベースをサーチするための方法であり、前記データベースは多数のクラスと、前記クラスの一つに関係する少なくとも一つのサーチ可能な属性と一対の前記クラス間の少なくとも一つのデータモデル関係とを持ち、

質問モデルを記述し、前記質問モデルは、第1のクラスと、前記第1のクラスに関係した少なくとも一つのサーチ可能な属性と、第1のデータモデルの関係により、前記第1のクラスに関係する第2のクラスと、前記第2のクラスに対する少なくとも一つのサーチ可能な属性とを持ち、

前記質問モデルを1組みのリレーショナルデータベースコマンド翻訳し、

前記リレーショナルデータベースコマンドをリレーショナルデータベースへ適用し、そしてそれに応答した質問結果を回復し、

前記質問結果を表示する、

ステップを含む方法。

(13) 前記リレーショナルデータベースは1組みの関係テーブルを含み、前記翻訳のステップは、

前記第1のクラスに関係した第1のリレーショナルデータベーステーブルと、前記第2のクラスに関係した第2のリレーショナルデータベースとを同一化し、

前記第1の関係テーブルと前記第2の関係テーブルとを結合して関係した結合

を得るために、前記第1のクラスと前記第2のクラスとの間のデータモデル関係に
に
応答して、少なくとも一つのデータベースコマンドを最初に発生し、

前記第1のサーチ可能な属性に関係した第1のコラムを同一視し、そして、

前記コラム内に前記関係結合の記録をテストするために少なくとも一つのデータベースコマンドを2番目に発生するステップを含む請求の範囲第12項記載の方法。

(14) 前記リレーショナルデータベースは1組みの関係テーブルを含み、前記翻訳のステップは、

前記第1のクラスに関係した第1のリレーショナルデータベーステーブルと、
前記第1のクラスのベースクラスに関係した第2のリレーショナルデータベースとを同一化し、

前記第1の関係テーブルと前記第2の関係テーブルとを結合して関係した結合を得るために、前記第1のクラスと前記第1のクラスのベースクラスとの間の継承関係に
に
応答して、少なくとも一つのデータベースコマンドを最初に発生し、

前記第1のサーチ可能な属性に関係した第1のコラムを同一視し、そして、

前記コラム内に前記関係結合の記録をテストするために少なくとも一つのデータベースコマンドを2番目に発生するステップを含む請求の範囲第12項記載の方法。

(15) 前記翻訳のステップは、前記質問モデルにより示された質問を実行するために1組みのデータベースコマンドを発生し、そして

前記リレーショナルデータベースのより早い質問を実行するために、前記組みのデータベースコマンドを最適化するステップを含む請求の範囲第12項記載の方法。

(16) データベースをサーチするための方法であり、前記データベースは多数のクラスと、前記クラスの一つに関係する少なくとも一つのサーチ可能な属性と
に
一対の前記クラス間の少なくとも一つのデータモデル関係とを持ち、

前記クラスの第1のリストを示し、

前記第1のリストからの第1のクラスの選択を受け取り、

前記第1のクラスと関係するサーチ可能な属性の選択を受け取り、

前記クラスの第2のリストを示し、前記第2のリストは、前記データモデル関係により、前記第1のクラスに關係したベースクラスを持つクラスを含み、

前記第2のリストから第2のクラスの選択を受け取り、

前記第2のクラスに關係したサーチ可能な属性の選択を受け取り、

1組みのリレーショナルデータベースコマンドを発生させ、前記組みのリレーショナルデータベースコマンドは、リレーショナルデータベースへの質問を呈するため有効であり、前記質問は、前記第1のクラスの第1の選択の語義上の効果、前記第1のサーチ可能な属性、前記第2のクラスおよび前記第2のサーチ可能な属性を持ち、

前記組みのリレーショナルデータベースコマンドを前記リレーショナルデータベースに適用し、そして、それらに回答して質問結果を回復し、

前記質問結果を表示する方法。

(17) 前記組みのリレーショナルデータベースコマンドは、前記データベース内のデータモデル関係に回答する少なくとも一つのテーブル結合を持つ請求の範囲第16項記載の方法。

(18) 前記組みのリレーショナルデータベースコマンドは、前記データベース内の継承関係に回答する少なくとも一つのテーブル結合を持つ請求の範囲第16項記載の方法。

(19) 前記データベースは、第1の測定ユニットに關係した少なくとも一つのサーチ可能な属性を含み、そして

前記組みのリレーショナルデータベースコマンドは、前記第1の測定ユニットから第2の測定ユニットへの変換に回答する少なくとも一つの比較を持つ請求の範囲第16項に記載の方法。

(20) 前記発生するステップは、前記データベースのメタモデルに回答し、前記メタモデルは、前記データベースのクラスに回答するオブジェクトの第1のクラスを含み、そして、前記データベースのサーチ可能な属性に回答するオブジェクトの第2のクラスを含む請求の範囲第16項記載の方法。

【発明の詳細な説明】

発明の名称

データベース構造に適したオブジェクトのモデリング、リレーショナルデータベース構造への翻訳、それらへの流動的なサーチ

発明の背景

1. 発明の分野

この発明は、データベース構造に適したオブジェクトのモデリングおよびリレーショナルデータベースの構造への翻訳およびそれらに対する流動的なサーチに関する。

2. 関連技術の記述

コンピュータのソフトウェアシステムの設計においては、各データアイテムをデータ種に関連付けることと、システムのすべてのエレメントに対する各データ種のオブジェクトへの比較的一様なインタフェイスを与えることが至上と考えられている。この技術は、システムのエレメントを、データ種、データ種への一定のインタフェイス、およびデータ種および他のデータ種の間の関係の属性に依存させることを可能にする。整数および浮動小数点のごとき、いわゆる”ビルトイン”のデータ種に加えて、この技術をより複雑なデータ種、ときには、ユーザーのシステムにより定義されたクラスを含む”クラス”と呼ばれものに拡張するのに優れている。(例えば、ユーザーは電話番号で呼ばれるクラスを定義してもよく、又、これにより、システムのエレメントが電話番号を、それらがあたかも基本的な情報の断片かのように、格納、操作または回収することを可能にする。)

ソフトウェアオブジェクトのクラスを定義する技術およびこれらのオブジェクトへのアクセスを制限することは、言語のプログラミングに適したオブジェクト(“OOP L”)と呼ばれる、C++およびスモルトークプログラミング言語のごときいくつかのプログラミング言語で現在一般である。

最近、ソフトウェアオブジェクトのクラスを定義すること及びこれらのオブジェクトを扱う技術は、データベースアプリケーションのために同様に有用であることが見いだされた。データベース(“OODB”)アプリケーションに適したオブジェクト

クトでは、ユーザーは、オブジェクト、これらのクラスの属性およびこれらのクラス間の関係を定義し、そして、これらのオブジェクトの例であるデータアイテムをデータベースに定住させる。（例えば、クラス電話番号が人々や会社およびコンピュータネットワークのための電話番号として格納するために使用されてもよい。）データベースの扱いに適したオブジェクトは、高速のアプリケーションおよびデータベースの開発および関連するソフトウェアの信頼性に利点を与えることが見いだされた。

しかしながら、この分野で提起された一つの問題は、多数のデータベースシステムが、別の異なったデータベース技術、“リレーショナルデータベース”処理(RDBM)を動作させるために設計されていることである。リレーショナルデータベースでは、データベースは、データアイテム間の関係を記した1組みのデータテーブルを備える。各テーブルは、1組みのレコードを含み、一つのレコードは各データアイテムに対するもので、各レコードは、1組みのコロンまたはフィールドを持ち、一つのコラムはデータアイテムに関連したデータ値を持つ。（例えば、人々のテーブル内の各レコードは、人々の電話番号に対して定義された一つのコラムを持ってもよく、同様に、会社のテーブル内の各レコードがその会社の電話番号のために定義されたコラムを持ってもよい。）リレーショナルデータベースは、大きいデータベースを扱う際に有用で効果的であることが見いだされている。それらの成功により、それらの利点が実質的にリレーショナルデータベースの処理システム、そしてアプリケーションおよび、リレーショナルデータベース処理システムをサポートもしくはインタフェイスするコンピュータプログラムに投資されている。

データベース処理に適したオブジェクトの利点を得ることは有利である一方、リレーショナルデータベース処理システムの効率性を維持し、かつ、インストールされた投資に有効である。

この分野で提起された一つの問題は、データベースに適したオブジェクトを創造し、操作し、扱うためのツールと一般に矛盾する。

発明者がSubodh BapatでRacal-Datacom社に譲渡された1994年3月1日発行の米

国特許5,291,583の“リレーショナルスキーマにおけるバシステントASN.1オブジェクトの自動格納”および発明者が同じSubodh BapatでRacal-Datacom社に譲渡された1994年3月15日発光の米国特許5,295,256の“リレーショナルスキーマにおけるバシステントオブジェクトの自動格納”は、C++またはスモールトークのごとき言語をプログラミングするのに適したオブジェクトを用いて決定された構造を、リレーショナルデータベース構造にコンパイルングするための方法を開示する。しかしながら、これらの特許に示された方法は、オブジェクトの自動格納に対するリレーショナルデータベース構造を創造する目的を達成するが、それは、一般に、オブジェクト体系およびオブジェクト関係を流動的に変形することを含み、データベース内のオブジェクトを流動的に組み込みエディットし、そして、データベースに対して流動的にサーチを発生させ実行する、データベースアプリケーションの要求された他のタスクに適切でない。

次の特許を当分野の例として示す

Consilium社に譲渡されたSubhash B.Tantry その他の発明者による米国特許5,398,336の“ファクトリー フロワー マネージメントに対するアーキテクチャーに適したオブジェクト”が1995年3月14日に請求されている。

インターナショナル ビジネス マシーンズ コーポレーションに譲渡されたRonald B.Baker その他の発明者による米国特許5,212,787の“メソッド アンド アパレイタス フォー アクセシング ア リレーショナル データベース ウイズアウト エキサイティング アン オブジェクト エンバイオロメント”が1993年3月18日に発布されている。

Xidak社に譲渡されたRobert N.Goldberg その他の発明者による米国特許5,201,046 “リレーショナルデータベース マネージメント システム アンド フォーストアリング リトリビング アンド モデファイニング ディレクティド グラフ データストラクチャ”が1993年4月6日に発布されている。

イーストマン コダック社に譲渡されたRobert M.Smith その他の発明者による米国特許5,181,162 “ドキュメント マネージメント アンド プロダクションシステム”が1993年1月19日に発布されている。

インタナショナル ビジネス マシーンズ コーポレーションに譲渡されたRobert L. Abraham その他の発明者による米国特許5,161,225 “バシステント フォー プロセッシング タイム コンシューミング アンド リューザブル クオーリーズ イン アン オブジェクティド データベース マネージメント システム” が1992年1月3日に発布されている。

ヒューレット パッカード社に譲渡されたTore J.M. Rishによる米国特許5,133,075 “メソッド オブ モニタリング チェインズ イン アトリビュート バリユー オブ オブジェクト イン アン オブジェクティド データベース” が1992年2月21日に発布されている。

インテリコープ社に譲渡されたFrederich N. Touその他の発明者による米国特許4,930,071 “メソッド オブ インテグレーティング ア ノレッジ・ベースド システム ウイズ アン アービトレイ データベースシステム” が1990年3月29日に発行されている。

従って、このリレーショナルデータベース・マネージメントシステムを使用できることを保持する一方、データベースアプリケーションに適したオブジェクトの利点を得られる、適したオブジェクトおよびリレーショナルデータベース構造の間でのコンパイルおよび翻訳の方法を提供するのが有利である。

発明の概要

本発明は、データベース構造に適したオブジェクトのモデリング、リレーショナルデータベース構造への翻訳およびそれらに対する流動的なサーチに対する方法およびシステムを提供する。本発明は、これにより、ユーザーがユーザーのオブジェクトデータベースを創造し、エディットし、扱うことを可能にし(方法およびシステムが流動的に1組みのリレーショナルデータベース構造に翻訳する)、前記データベースに対するオブジェクトを創造し、エディットし扱うことを可能にし(方法およびシステムが流動的にこれらのリレーショナルデータベース構造のためのデータに翻訳する)、そして、そのオブジェクトデータベースに適用される質問を創造し、エディットし扱う(方法およびシステムがこれらのリレーショナルデータベース構造に適用される質問に流動的に翻訳する)ことを可能にする。

この方法およびシステムは、ユーザーのオブジェクトデータベースのメタモデルを含み、それ自身はオブジェクトデータベースであり、それ自身が、リレーショナルデータベースエンジンにより、扱いのために1組みのリレーショナルデータベースに翻訳される。そのメタモデルは、1組みのクラス、オブジェクトおよびクラス間の関係を含み、クラスおよびシステムのクラス間の関係をモデルする。これらのクラスの各々は、1組みのサーチ可能な属性を含み、これらの関係の各々は、継承の関係(ベースクラスと派生のクラスとの間)もしくは、データモデルの関係(一つから一つへの、一つから多くへのもしくは多くから多くの)を含んでもよい。ユーザーのオブジェクトデータベースのデータモデルは、メタモデル内の実際のオブジェクトによりモデリングされ、そして、ユーザーのオブジェクトデータベースのエディットもしくは扱いは、メタモデル内のオブジェクトを創造し、変更し、削除することによりモデリングされる。メタモデルは又、それがユーザーのオブジェクトデータベースをモデリングするがごとく、自身をモデリングし、そして、ユーザーのオブジェクトデータベースのごとく同じ方法で扱われてもよい。

好ましい実施では、ユーザーのオブジェクトデータベース内の各クラスは、リレーショナルデータベースのテーブルによりモデリングされ、各サーチ可能な前記クラスの属性は、前記テーブル内のコラムによってモデリングされ、そして、前記クラスの各オブジェクトは、前記テーブル内の列、および前記テーブルのためのベースクラスを表すテーブル内の対応する列によりモデリングされる。

このような各テーブルは、システムにより発生された“オブジェクトID”を含み、それはシステム内の各オブジェクトに対する個々のIDを含む。二つのオブジェクト間の各関係は、第1のオブジェクトから第2のオブジェクトへのポインターを与えることによってモデリングされ、そのポインターは、第2のオブジェクトのオブジェクトIDを有する第1のオブジェクトのテーブル内のコラム(これは関係をモデリングするシステムにより発生される)を含む。これらのリレーショナルデータベース構造は、ユーザーのオブジェクトデータベースを創造し、エディットし、扱うユーザーコマンドに応答して流動的にもしくは増加的に、
創

造され、モデリングされ、もしくは削除される。

少なくとも二つのタイプの関係は、システム、継承関係およびデータモデルの関係によりモデリングされる。継承の関係は、ベースクラスと派生クラスの間には存在する。派生クラスのオブジェクトメンバーは、ベーステーブルのためのテーブルおよび派生クラスのためのテーブルの双方に入り込み、両テーブル内に同じオブジェクトIDが与えられる。二つのクラス間の継承関係は、オブジェクトIDコラム上の二つの両テーブル間のリレーショナルデータベースJOINによりモデリングされる。複数の継承のために、同じオブジェクトIDを持つ一つのオブジェクトは、派生クラスと多数の各ベースクラスとに入り込む。

データモデルの関係は、ユーザーのオブジェクトデータベースに対するユーザーにより創造され、一つから一つ、多くから一つ、多くから多くへの関係を含んでもよい。一つから一つおよび多くから一つへの関係は、第1のオブジェクトから第2のオブジェクトへのポインターを与えることによりモデリングされる。多くから多くの関係は、クロスリンクのクラスを与えることによりモデリングされ、多くから多くへの関係における第1のオブジェクトのオブジェクトIDを持つ第1のクラスのための一つのコラムを持つテーブルを用いて実行され、第2のクラスに対する一つのコラムは多くから多くへの関係における第2のオブジェクトのオブジェクトIDを持ち、そして、実際に関係するオブジェクトの各組みに対する一つの列を持つ。一つから一つへと多くから多くへの関係は、データモデルの関係をモデリングするコラム上の二つの対応するテーブルの間のリレーショナルデータベースJOINにより、モデリングされ、多くから多くの関係は、データモデル関係をモデリングする二つの対応するテーブルとクロスリンクテーブルとの間のリレーショナルデータベースJOINにより、モデリングされる。

本方法およびシステムは、多数のクラスのオブジェクトを縦方向のサーチを持つユーザーのオブジェクトデータテーブルを質問することを提供する質問モデルを含む。ユーザーは、サーチされるべき一つまたはより多くのクラス、これらのクラスにおけるサーチ可能なオブジェクトの属性(これらのサーチ可能な属性のための特殊な値を持つように)における制限、およびこれらのクラス内のオブジェクト

クトのために示された情報を選択する。このサーチ記述に応答して、システムは、継承関係を実行するのに必要ないかなるコマンドおよびユーザーのオブジェクトデータテーブルのデータモデル関係を自動的に含むユーザーのオブジェクトデータテーブルに対応するリレーショナルデータベース構造に適用されるリレーショナルデータベースの質問(好ましくは最適化されたSQLコマンドを含む)を発生する。好ましい実施例は、サーチ可能な属性の値に対する自動のユニット変換を提供する。リレーショナルデータベースの質問がリレーショナルデータベースに適用されたとき、システムは、リレーショナルデータベースから供給された情報を受信し、その情報をサーチされたクラスの表示属性に基づきユーザーに示す。

好ましい実施では、ユーザーのオブジェクトデータベースの態様へのアクセスコントロールをモデリングするために、メタモデルは、第1の組みの予め決定されたクラスを含む。アクセスは、選択されたユーザーもしくは選択されたユーザーグループに制限されてもよく、そして、個々のオブジェクト、オブジェクトのクラスまたはこれらのオブジェクトの属性のために特定化されてもよい。好ましい実施例では、メタモデルは又、ユーザーのオブジェクトデータベースにおけるオブジェクトへのグラフィックおよびテキストオブジェクトのアタッチメントをモデリングするために第2の組みの予め決定されたクラスを含む。グラフィックおよびテキストオブジェクトは、ユーザーのオブジェクトデータベース内のオブジェクトと結合してもよく、又、メタモデルが埋め込まれたオペレーションシステムにより維持されたファイルおよび他のオブジェクトと結合してもよい。

図面の簡単な説明

図1は、ユーザーのオブジェクトデータベースの例を示したデータモデルのダイヤグラムを示す。

図2は、データベース構造に適したオブジェクト、リレーショナルデータベースへの翻訳およびそれらへの流動的なサーチのモデリングのためのシステムのブロックダイヤグラムを示す。

図3は、ユーザーのオブジェクトデータベースをモデリングするためのメタモデルのデータモデルのダイヤグラムを示す。

図4は、ユーザーのオブジェクトデータベースのためのアクセスコントロールをモデリングするための1組みのクラスのデータモデルダイアグラムを示す。

図5は、ユーザーのオブジェクトデータベースのための好ましいグラフィック情報の表示のための1組みのクラスのデータモデルダイアグラムを示す。

図6は、ユーザーのオブジェクトデータベースのデータモデルを構築しエディットするための工程のフローダイアグラムを示す。

図7は、適したオブジェクトとリレーショナルデータベース構造との間の翻訳のための工程のフローダイアグラムを示す。

図8は、図1のサンプルのユーザーのオブジェクトデータベースに対応するユーザーのリレーショナルデータベースのデータモデルダイアグラムを示す。

図9は、図9Aおよび図9Bに関する。図9Aは、オブジェクトデータベースの縦方向のサーチおよび、適したオブジェクトとリレーショナルデータベース構造との間のサーチ翻訳のための工程のフローダイアグラムを示す。図9Bは、縦方向のサーチのための工程で使用されたデータ構造のブロックダイアグラムを示す。

好ましい実施例の記述

次の記載では、好ましい工程ステップ、データ構造およびユーザーインタフェースに関して本発明の好ましい実施例が記述される。しかしながら当業者においては、この実施例の熟読後は、本発明の実施例は、一般ユースのコンピュータまたは1組みの一般ユースのコンピュータ、プログラム制御下の動作、および一般ユースのコンピュータの変形または1組みの一般ユースのコンピュータ、データ構造およびここで記述したユーザーインタフェースを用いて実施できることが理解されよう。

ユーザーのオブジェクトデータベース例

図1は、ユーザーのオブジェクトデータベースの例のデータモデルダイアグラムを示す。この例は、オブジェクトデータベース、オブジェクトデータベースのリレーショナルデータベースへの翻訳およびオブジェクトデータベース上で実行されるデータベース動作の例をサポートするために提供される。

ユーザーのオブジェクトデータベース100の例は、1組みのクラス101を含み、各々は1組みのサーチ可能な属性102を含み、そしてその各々は、格納し、扱いもしくは回収するためのデータベースに適したオブジェクトのためのデータ種を示す。そのクラス101は、体系(より詳しくは指図された周期的なグラフに)を形成し、各クラス101は、ゼロまたはより多くの派生クラス101のためのベースクラスとなる。派生クラス101がベースクラス101から派生したとき、そのクラスは、ベースクラス101のサーチ可能な属性102のすべてを継承し、そして、派生クラス101に特別の付加的なサーチ可能な属性102を持ってもよい。

この例では、オブジェクトデータベース100は、クラス101 コンポーネントを含み、オブジェクトデータベース100の語義は、電気回路コンポーネントに決定される。各オブジェクトのクラス101 コンポーネントは、サーチ可能な属性102“名前”を持ち、オブジェクトデータベース100の語義は前記コンポーネントの名前で定義される。

この例では、クラス101 コンポーネントは、ベースクラスであり、それから二つのクラス101、アナログコンポーネントおよびメモリが派生される。これらの二つのクラス101がクラス101 コンポーネントから派生されるので、それらのクラスは、クラス101 コンポーネントのすべての属性を継承し、クラス101の各オブジェクト、アナログコンポーネントおよびクラス101のオブジェクトクラスも又、サーチ可能な属性102“名前”を持つ。この例では、クラス101の各オブジェクトは、付加的なサーチ可能な属性102“サイズ”を持つ。

クラス101 メモリも又、ベースクラスであり、そのベースクラスから二つのクラス101 流動的(ダイナミック)メモリおよびスタティックメモリが派生する。従ってそれらは、クラス101 メモリ、そしてそれ故、クラス101 コンポーネントのすべての属性を継承し、そのため、クラス101 スタティックメモリの各オブジェクトは、サーチ可能な属性102のように“名前”および“サイズ”を持つ。

クラス101の別の例は、メーカーである。この例では、各クラス101 メーカーの各オブジェクトは、“名前”と“都市”をサーチ可能な属性102として持つ。この例では、クラス101 メーカーは、ベースクラスであり、そこからクラス101 国内メーカーおよび外国メーカーが派生する。クラス101 外国メーカーは又、サーチ可能な属性102 “国名”を持つ。

この例では、クラス101 コンポーネントは、クラス101 メーカーを有するデータモデル関係103を持つ。この関係103は、多くから一つであり、クラス101 コンポーネントの各オブジェクトは、クラス101 メーカーの関係したオブジェクトを持ち、クラス101 メーカーの各オブジェクトは、ゼロまたはより多くのクラス101 コンポーネントの関係した要素を持つ。

サーチ可能な属性102に加えて、派生したクラス101の各オブジェクトは、そのベースクラス101の関係を継承する。これにより、クラス101 スタティックメモリの各オブジェクトは、クラス101 メモリの関係103を継承し、前記メモリはクラス101 コンポーネントの関係103を継承する。クラス101 コンポーネントの関係103のひとつは、クラス101 メーカーを有する関係103であり、そのため、各スタティックメモリはメーカーを持つ。同様に、各外国のメーカーは、スタティックメモリ、流動的メモリおよびアナログコンポーネントを作ってもよい。

当業者であれば、この適用を精読した後でこのユーザーのオブジェクトデータベースの例で多くの変形およびこのような変形に対してここに開示された方法およびシステムの容易な適用を容易に認識できるであろう。ここに開示された方法およびシステムの可能性は、別の発明や経験を必要としないであろう。

コンパイルおよび翻訳のためのシステム

図2は、データベース構造に適したオブジェクトおよびリレーショナルデータベース構造への翻訳をモデリングするためのシステムのブロックダイアグラムを示す。

データベース構造およびリレーショナルデータベース構造への翻訳のモデリングのためのシステム200は、ユーザー201からのコマンドおよび記述を受信

するためにユーザインタフェイス210を含む。ここで示されたように、ユーザ201により供給されたコマンドおよび記述は、ユーザのオブジェクトデータベース100に関する記述情報を含む。このシステム200は、ユーザのオブジェクトデータベース100に関する記述情報を受信し、その情報をユーザデータベースモデル200を提示するためにメタモデル230内に格納する。メタモデル220およびユーザのデータベースモデル230は、リレーショナルデータベース内の構造により、それら自身により提示され、そのリレーショナルデータベース250はリレーショナルデータベース240を用いて実行される。

好ましい実施例では、ユーザインタフェイス210は、グラフィックユーザインタフェイス211(“GUI”)を含み、それは創造するためのエディット用ツールを持ち、ユーザのデータベースモデル230をエディットする。グラフィックユーザインタフェイスはこの分野では公知であり、当業者であれば、この明細書を精読した後に、GUI211に要求される機能を実行するために公知のコンピュータシステムを変更することは簡単であり、別の発明や経験を必要としないことが理解されよう。

好ましい実施例では、ユーザインタフェイス210は又、ユーザのオブジェクトデータベース100に関する記述情報を入力するための別のツールを含む。これらの別のツールは、テキスト記述ツール212を含み、それは記述情報を記録するためのテキストフォーマットを持ち、ファイルまたは他のソフトウェアオブジェクトから前記テキストフォーマットを読み出すために配置される。好ましくは、他のソースからシステム200ヘデータの容易かつ迅速な翻訳を容易にするために、テキストフォーマットは、他のデータベースのごとき他のプログラムで記載された物とコンパチブルである。

これらの別のツールは又、ユーザのオブジェクトデータベース100のコマンドおよび記述をシステム200に供給するために、アプリケーションプログラミング用インタフェイス213(“API”)を含み、これはユーザ201により呼び出されたプログラムとのインタフェイスを供給するための1組みのプログラミング構成を持ち、そして、ユーザのデータベースモデル220を創造しエ

ディットするために使用するプログラム呼び出しとデータを受信するために配置される。好ましくはAPI 213は、GUI 211と同様な記述的および命令的なパワーを持ち、そのため、ユーザー201は、GUI 211と同じ機能を実行するために別のプログラムを呼び出してもよい。

その機能は、GUI 211により供給され、テキスト記述212およびAPI 213は図6で更に記述される。

メタモデル220は、オブジェクトデータベースを持ち、そのデータベースはオブジェクトとして、ユーザーのオブジェクトデータベース100内のクラス101、これらのクラスのサーチ可能な属性102およびこれらのクラス間の関係103を持つ。メタモデル220は、図23で更に記述される。

好ましい実施例では、リレーショナルデータベースエンジン240は、カリフォルニア、レッドウッドショアーズのオラクル コーポレイションで販売の“オラクル”のごとき、標準のリレーショナルデータベースの処理ツールを含む。別の実施では、リレーショナルデータベースエンジン240は、“SQL”データベース処理言語、あるいは同様な記述的および命令的なパワーを持つ別のデータベース処理言語におけるリレーショナルデータベースコマンドを受ける製品を含んでもよい。

好ましい実施例では、リレーショナルデータベースエンジン240は、クライアント/サーバー、マルチユーザー、マルチユーザーリレーショナルデータベースエンジンを使用するネットワークされたアーキテクチャーを含み、そのため、(1)多数のユーザー201がリレーショナルデータベース250を同時に操作してもよく、(2)要求はリレーショナルデータベースエンジン240により処理され、そして(3)多数のユーザー201およびシステム200の多数のコピーが多数のリレーショナルデータベース240に結合されたネットワークに配置されてもよい。

ユーザーデータベースの構築

ユーザーインタフェース210は、ユーザーのオブジェクトデータベース100に関するコマンドおよび記述をユーザー201から受信し、それに応答して、

ユーザーデータベースモデル230を構築しエディットする。ユーザーデータベースモデル230がメタモデル220内に1組みのオブジェクトを含むので、ユーザーデータベースモデル230を構築しエディットする工程は、単にメタモデル220内に構築およびエディット用のオブジェクトを含む。これらのオブジェクトは継承オブジェクトとしてシステム200により維持され、そのため、ユーザーデータベースモデル230は、システム200により、メタモデル220の継承部分として維持される。

メタモデル220自身は、メタモデルリレーショナルデータベース221により示され、そのデータベースは1組みのリレーショナルデータベース、これらのテーブルの属性、これらのテーブル内のコラム、これらのテーブルの一次および二次のキー、およびメタモデルリレーショナルデータベース221の他の決定用特徴を含む。ユーザーデータベースモデル230は、メタモデルリレーショナルデータベース221内のオブジェクトにより示され、それらのテーブル内に列、これらの列に対するコラムに入った特定値、および一つのテーブルの1列から関連するテーブルの1列へのポインターを含む。ユーザーデータベースモデル230を構築しエディットする工程は、つまり、メタモデル220内のオブジェクトを構築しエディットすることは、単に構築およびエディット用クラス101、オブジェクト、サーチ可能な属性102および、ユーザーのオブジェクトデータベース100に関する記述情報を示すメタモデルリレーショナルデータベース221内の列、値およびポインターにより示されたクラス101間の関係を含む。

メタモデル220は又、メタモデルリレーショナルデータベース221により示されるので、メタモデルリレーショナルデータベース221内の1組みのオブジェクトは、メタモデル220自身を示す。メタモデル220を示しているメタモデルリレーショナルデータベース221内のこれらのオブジェクトは、ユーザーデータベースモデル230を示すオブジェクトのようにエディットされてもよい。事実、ユーザーデータベースモデル230は、メタモデル220内のオブジェクトをエディットすることにより、単に構築しエディットされてもよい。メタモデル220の構造でさえメタモデル220内のオブジェクトをエディットするこ

とにより、削除されてもよい。

トリガー用のイベントに応答して、システム200は、ユーザーリレーショナルデータベース231内のリレーショナルデータベースを記載し創造するために1組みのSQLコマンドを使用してユーザーモデル230をユーザーリレーショナルデータベース231に翻訳する。これらのSQLコマンド232は、リレーショナルデータベーステーブル、これらのテーブルの属性、これらのテーブル内のコラム、これらのテーブルの一次および2次のキーおよびユーザーリレーショナルデータベース231の他の決定用属性を記載する。これらのSQLコマンド232は、リレーショナルデータベースエンジン240へ翻訳され、その翻訳されたものはその後、記載し、ユーザーリレーショナルデータベース231を創造する。

好ましい実施では、ユーザーデータベースモデル230の翻訳のためのトリガー用イベントは、ユーザーリレーショナルデータベース231を創造するために、ユーザー201によるGUI211へのコマンド、もしくは、ユーザー201により呼び出されたAPI213へのプログラムによるコマンドを含む。しかしながら、これとは別に時限化された自動セーブのごときトリガー用イベントまたは、ユーザーデータベースモデル230内の十分な変化も又、システム200により認識されてもよい。ユーザーデータベースモデル230がテキスト記述ツール212を使用して記載されたとき、トリガー用イベントは、テキストファイル自身内にトリガー用のコマンド、テキストの翻訳をさらに持つかといったファイル状態のエンド又は別の状態を含む。

適したオブジェクトおよびリレーショナルデータベース構造との間の翻訳に対する工程は、図7で更に記載される。

ユーザーデータベースの取り込み

ユーザーインタフェイス210は又、ユーザー201から、ユーザーのオブジェクトデータベース100のためのオブジェクトに対応する、ユーザーデータベースモデル230のための取り込みオブジェクトに関するコマンドおよび記述を受け取る。これに응答して、システム200は、ユーザーリレーショナルデータベース

ース232のためと、その次のユーザーデータベースモデル230内の記載のために、データアイテムを構築しエディットする。

システム200は、リレーショナルデータベース231のテーブル内の列を挿入し、変形し、または削除するために、ユーザーデータベースモデル230のためのオブジェクトを1組みのSQLコマンド233の中へ移動する。これらのSQLコマンド233は、リレーショナルデータベース240へ送出され、このエンジンは、これらのテーブル、これらの列に対するコラムのための特定値および一つのテーブルから関連テーブルへのポインターを生成する。

好ましい実施例では、ユーザー201は、ユーザーデータベースモデル内の各クラス101に対する完全なチェックを記載してもよく、その結果、オブジェクトがユーザーデータベース231に加えられたとき、完全なチェックが実行され、そのオブジェクトは完全なチェックに迎合することが確実にされる。例えば、図1に示されたユーザーのオブジェクトデータベース100の例では、ユーザー201は、各クラス101、コマンドのオブジェクトは、一つとクラス101、メーカーのオブジェクトに関係した一つを持つ。この例では、ユーザー201は、クラス101、コンポーネントとクラス101、メーカーとの間のデータモデル関係103内のこの完全チェックを記述してもよい。

同様に、ユーザー201は、サーチ可能な属性102のために単に正当な値として、1組みの数えられた値または値の範囲を記載してもよい。ユーザー201がそのように記載したとき、システム200は、図3に関して更に述べるように、オブジェクトを創造し、確定した1組みの数えられた値または確定した値の範囲を述べ、そして、前記オブジェクトをサーチ可能な属性102と関係させる。好ましい実施例では、ユーザーインタフェイス210は、サーチ可能な属性102のために、特定のオブジェクトのためのサーチ可能な属性102のための値を入力する前に、確定した組みの数えられた値または確定した値の範囲を示すために、ユーザー201にコマンドを与える。このことは、新しい値を持つオブジェクトを取り込む前に、ユーザー201がどの値がサーチ可能な属性102に対して適切かを決定することを可能にする。ユーザー201がオブジェクトの他のサ

ーチ可能な属性102のために値を既に入力したとき、ユーザーインタフェイス210は、これらの他のサーチ可能な属性102のための同じ値を持つオブジェクトのためにデータベース内に既にあるこれらの値のみを記載する。例えば、図1に示されたユーザーのオブジェクトデータベース100の例では、もしユーザー201がクラス101、メモリのオブジェクトを作成し、クラス101、メーカーの関係したオブジェクトのためにサーチ可能な属性102“名前”に対する値を既に入力し、そして、サーチ可能な属性102“サイズ”に対する値を入力しているところならば、ユーザー201は、前記メーカーと共に既にメモリに存在する“サイズ”に対するすべての値を表示するために、ユーザーインタフェイス210をリクエストしてもよい。

完全なチェックに加えて、ユーザー201は、特定のクラス101のオブジェクトが生成されるか破壊されたとき、“ビジネスルール”、つまり拡張機能が実行されるために記載してもよい。例えば、ユーザー201は、クラス101、コンポーネントのオブジェクトのために、前記クラス101の新しいオブジェクトが作成される毎にシステム200が新規なシリアルナンバーを生成する、ビジネスルールを用いてサーチ可能な属性“シリアルナンバー”を記載してもよい。クラス101のための拡張機能(クラス機能)およびサーチ可能な属性102(クラス属性機能)は、更に図3に関して述べる。

ユーザーデータベースへの質問

ユーザーインタフェイス210は又、ユーザーのオブジェクトデータベース100への質問およびサーチ(つまり、ユーザーのリレーショナルデータベース231内のオブジェクトに質問しサーチする)に関するコマンドおよび記述をユーザー201から受け取る。それに応答して、システム200は、ユーザーリレーショナルデータベース231に与えられるべき質問のために、質問モデル260を構築しエディットする。質問モデル260を構築しエディットする工程は、図9に関して述べる。

トリガー用イベントに応答して、システム200は、ユーザーのリレーショナルデータベース231に質問(グループ化および並べ替えのごときSQL質問の

特徴を含む)するために質問モデル260を1組みのSQLコマンド261へ翻訳する。これらのSQLコマンド261は、リレーショナルデータベースエンジン240へ送信され、そのエンジンは、その質問をユーザーリレーショナルデータベース231に与え、その質問から1組みの質問結果251を発生させ、そして、ユーザー201に示すために1組みの質問結果251を戻す。

好ましい実施例では、質問モデル260の翻訳のためのトリガー用イベントは、質問結果251を創造するために、ユーザー201によるGUI211へのコマンド、テキストコマンドによるテキスト記述ツール212へのコマンドおよびユーザー201により呼び出されたプログラムによるAPI213へのコマンドを含む。質問モデル260は又、ユーザー201への質問の結果を示すことに関する記述情報を含み、その情報は、質問のために、システム200により、SQLコマンド261内へ組み込まれる。

質問モデル260およびリレーショナルデータベースの質問への質問モデル260の翻訳は図9に関して更に述べる。

ユーザーのオブジェクトデータベースのためのメタモデル

図3は、ユーザーのオブジェクトデータベースをモデリングするためのメタモデルのモデルダイアグラムを示す。

メタモデル220は、1組みのクラス101、各クラス101に対する1組みのサーチ可能な属性102、クラス101間の1組みの関係103を含み、これにより、ユーザーのオブジェクトデータベース100についての記録情報のためのシステムオブジェクトデータベースを形成する。

図3において各ボックスはクラス101を示し、ボックス間の各ラインはクラス101間の関係103を示す。

クラス101と関係103との接続点で、シンボルは関係103が結合している、クラス101のオブジェクトの個数を示し、個数はゼロ、一つ、1より多くがある。シンボルは二つのシンボルパーツを持ち、円シンボルのパーツは0のオブジェクトを示し、1本のラインシンボルのパーツは一つのオブジェクトを示し、フォーク状のラインシンボルのパーツは一つより多くのオブジェクトを示す。

こ

れにより、一つのシンボルは、例えば、第2のクラス101内の一つまたはより多くのオブジェクトに対応する第1のクラス101における各オブジェクトを示す。

太字のボックスは、クロスリンクのクラス101を示し、その目的は、オブジェクト間に多くから多くの関係のために備えられる。このような各クラス101の各々は、二つまたはより多くの関係103を持ち、少なくとも、第1のクラス101への第1の多くから一つへの関係103と第2のクラス101への多くから一つへの関係103である。これにより、クロスリンクのクラス101内の各オブジェクトは、少なくとも注文された一对の要素、第1のクラスからの一つのオブジェクトおよび第2のクラス101からの一つのオブジェクトを含み、二つのオブジェクト間の一つから一つの関係を示す。

メタモデルクラス

好ましい実施例では、メタモデル220は、テーブル3-1に示したクラスを含む。

テーブル3-1

<u>クラス</u>	<u>クラス内のオブジェクトの記載</u>
クラス	メタモデルによりモデリングされたクラス
クラスリンク	親／子のクラス間のクラスリンク
ポインタークラスの属性	ポインターにより別のクラスにモデリングされたサーチ可能な属性
確定したクラスのポインター	ポインティングオブジェクトとターゲットのクラスとの間の関係
クラス属性	メタモデルによりモデリングされたサーチ可能な属性で内蔵されたタイプの値を持つ
属性グループ	関連するグループの属性
論理データタイプの定義	属性に対する論理データタイプ
数えられた確定値	属性に対する組みの数えられた確定値
確定した値の範囲	属性に対する確定した値の範囲

測定値のユニット

ユニット変換公式

クラス属性機能動作

クラス機能動作

カスタム機能定義

外部アクセス機能動作

公式

形態

サーチパネルの形態

パネル結果の形態

ユーザーのオブジェクトデータベース100は1組みのクラス101を含む。
メタモデル220では、ユーザーのオブジェクトデータベース100の各クラスは、クラスクラス301のオブジェクトによりモデリングされる。

好ましい実施では、クラスクラス301の各オブジェクトはテーブル3-2内の情報を含む。

テーブル3-2

属性

オブジェクトID

クラスレベル

クラスDB名

クラスグループ

記述

このオブジェクトに対するユニークなID

クラスに対して表示されるべき名前

ユーザーリレーショナルデータベースで使用されるべきクライに対するユニークな名前

、クラスが、メタモデル、グラフィックや許可

のごとく、システムにより与えられた予め決定のクラスまたはユーザークラスの一部であるかの指示

クラスタイプ	クラスがメタモデル、クロスリンククラス、一つの“フォルダー”名前下の派生クラスをグループ化する一つの目的のために創造されたベースクラスの一部かの指示
システムの記述か	このクラスがシステムの一部であるかの真相
オブジェクトのナンバー	このクラスの現在のオブジェクトのナンバー
D B ブロックのナンバー	このクラスにより現在使用されているデータベースブロックのナンバー
オブジェクトの最大 #	クラスに対して期待されるオブジェクトの最大ナンバー
クラスの記述	ユーザーの便宜のためのクラスの記述
未決定の動作	クラスに対する次の未決定の動作の指示、例えばリレーショナルテーブルの創造、変形、リレーショナルテーブル、リレーショナルテーブルの再創造、または動作無しの未決定
動作	このクラスのためにとられた動作の指示、例えば、“未決定の動作”の実行、表示器の形態の変更、オラクル的な“視野”のリフレッシュ、クラス内のオブジェクトの個数の計数または行われた動作の非実行である。

属性 1 0 2 “クラス D B 名” は、システムにより発生され、そして、好ましくは、例えばある長さまたは単にあるキャラクターを持つように制限されるがごとく、リレーショナルデータベースエンジン 2 4 0 により課せられた名前の制限を満足させることが要求される。S Q L コマンド 2 3 2、2 3 3 および 2 6 1 のごとく、このクラス 1 0 1 に対応するリレーショナル構造をアクセスするための S Q L コマンドを発生させるためのシステムにより、データベース “クラス D B 名” がシステム 2 0 0 により使用される。好ましい実施例では、ユーザー 2 0 1 は、システム 2 0 0 により発生された “クラス D B 名” を無効にしてもよいが、リレーショナルデータベースエンジン 2 4 0 により課せられた名前の制限を満足し

なくてはならない。

属性102、“クラスグループ”、“クラスタイプ”および“システムの記述か”は、システム200内のクラス101の目的についての情報を維持する。もし、ユーザー201が、メタモデル220の一部である場合のように、システム200の動作に批判的であるクラス101をモデリングするならば、システム200は、ユーザー201に警告し、動作が実際に行われようとする確信を要求する。

属性102、“オブジェクトの最大#”は、ユーザー201により与えられ、この属性102はいかなる種類の難しい制限を課すことはないが、ユーザー201による提案であり、そして、オブジェクトの個数がこの属性の値を超過したとき、システム200は、好ましくはデフォルト値および拡張のための手順を与える。その属性102、“オブジェクトの個数”および“DBブロックの個数”はシステム200により与えられる。これらの属性102は、図9に関して示されるように、リレーショナルデータベースエンジン220を用いてテーブルを創造しサーチする際、システムの効率を改善するために使用される。

属性102、“未決定の動作”および“動作”は、リレーショナルデータベース250内のその表示に関するクラス101の現状についての情報を維持する。システム200を同時に使用する一つ以上のユーザー201があるならば(これがシステム200の好ましい動作)、一つより多いユーザー201が同じクラス101を同時に変形しようと試みる。そのシステム200は、いかなるクラス101も矛盾状態に入ることがないようにする試みが確実となるように、これらの属性102を使用する。データベースの当業者には、データベースシステムの同時のアップデートにも拘わらず、データベースを完全に維持するための公知の多くのテクニックがあり、この明細書を精読した後、システム200へのこのような技術の適用は、別の発明や高度な経験を必要としないことが理解されよう。

親／子のクラス

ユーザーのオブジェクトデータベース100では、各クラスはゼロまたはより多くの親クラス(ベースクラスから派生したもの)と、ゼロまたはより多くの子ク

ラス(ベースクラスから派生)を含む。メタモデル220では、クラスクラス301の各オブジェクトは、ゼロまたはより多くの親オブジェクトのクラスクラス301を含んでもよく、又、ゼロまたはより多くの子オブジェクトのクラスクラス301を含んでもよい。これらの多くから多くの親／子の関係103が、クロスリンククラスリンク302を用いて示される。

クラスクラスリンク302内の各オブジェクトは、子クラス101を親クラス101にリンクする。子クラス101は多数の継承に基づき、一つ以上の親クラス101を持ってもよい。

好ましい実施例では、クラスクラスリンク302の各オブジェクトはテーブル3-3の情報を含む。

テーブル3-3

<u>属性</u>	<u>記述</u>
オブジェクトID	このオブジェクトに対するユニークなID
親クラス	親クラスに対するオブジェクトID
子クラス	子クラスに対するオブジェクトID
分配されたDBノード	分配されたデータベース内のノード、そのデータベースに対してこのオブジェクトがリンク

サーチ可能な属性102“分配されたDBノード”は、分配されたデータベース内でノードを確認する。クラスクラスリンク302のオブジェクトが分配されたデータベース内の別のノードとリンクしたとき、サーチ可能な属性102、前記オブジェクトに対する“分配されたDBノード”は別のノードを確認し、そしてシステム200は、リンクされたノードにて関連する工程を生み出すことにより、前記リンクに応答する。これにより、ユーザー201は、リレーショナルデ

ータベースJOIN動作、および、分配されたリレーショナルデータベース250の多数のノードに対して縦方向のサーチを実行できる。

ユーザー250が派生したクラス101を創造し、またはクラスのベース／派生状態を変えると、システム200は、ベース／派生の関係をモデリングするために、クラスクラスリンク302のオブジェクトを創造しエディットする。

サーチ可能な属性

ユーザーのオブジェクトデータベース100は、各クラスに対して1組みのサーチ可能な属性を含む。サーチ可能な属性は、整数および浮動小数点のごとき、“組み込まれた”データタイプであるデータ値を含むか、又は、ユーザーで決定されたクラス内のオブジェクトのごとく示される、自身がユーザー決定のデータタイプであるデータ値を含んでもよい。メタモデル220では、組み込みデータ種を含むタイプサーチ可能な属性は、クラスクラス属性303により示され、一方、ユーザー定義のデータ種を含むサーチ可能な属性は(ユーザー定義のデータ種のオブジェクトへのポインターにより実行される)、クラス101 ポインタークラス属性304により示される。

クラスクラス301は、クラス101をユーザー定義のデータ種を持つサーチ可能な属性にリンクする、クラスポインタークラス属性304への関係103を持つ。クラスポインタークラス属性304は、クラスポインター確定クラス305への関係を持ち、そして、そのクラスポインター確定クラス305は、クラス301への関係を持ち、ユーザー定義のデータ種を持つサーチ可能な属性102を前記データ種を決定するクラス101にリンクする。

サーチ可能な属性：ユーザー定義のデータベース

クラスポインタークラス属性304は、ユーザー定義のデータ種を持つクラス301のいずれかのオブジェクトに対して、各サーチ可能な属性に対するひとつのオブジェクトを含み、クラス301の各オブジェクトは、一つまたはより多くの、ポインタークラス属性304のオブジェクトを持つ。クラスポインタークラス属性304の各オブジェクトは、ユーザー定義のタイプへのポインターを持ち、そのポインターはクラスのオブジェクトポインター確定ク

ラス305のオブジェクトである。

好ましい実施では、クラスポインタークラス属性304の各オブジェクトはテーブル3-4の情報を持つ。

テーブル 3-4

属性記述

オブジェクトID	このオブジェクトに対するユニークなID
ポインター属性ラベル	属性のために表示されるべき名前
ポインター属性DB名	ユーザーリレーショナルデータベース内で使用されるべき属性に対するユーザーな名前
表示命令	クラス内でこの属性を他のものと比較して表示する命令
ターゲットにより所有されるか	ポイントしたクラスから許可を継承するかの真相
値が必要とされるか	値が無効でないかの真相
コラムのインデックスか	この属性に対するコラムがユーザーリレーショナルデータベース内にインデックされたかの真相
コラムのインデックスタイプ	リレーショナルデータベース内のインデックスのタイプ
外部よりの派生か	システムにより発生されたかの真相
リレーショナルの意味	ユーザーリレーショナルデータベース内の属性のリレーショナルの意味、
情報の形態	いつこの属性が表示されたかの指示：サーチに対する、サーチ結果に対する、そられの双方に対する、またはどちらでもない
アップデートルールの挿入	いつこの属性を持つ新しいオブジェクトを挿入またはアップデートしたかを追跡するルール、 例えば、関連したオブジェクト、いかなる関連したオブジェクトへの縦のアップデートまたは、関連したオブジェクトのアップデート上で無効にするためのセットポインター
ルールを削除	いつこの属性を持つオブジェクトを削除したかを追跡するルール、例えば、関連するオブジェ

	クト、いずれかの関連するオブジェクトへの縦方向の削除、関連するオブジェクトの削除を無効にするセットポインターまたは関連するオブジェクトの削除における値をデフォルトにするセットポインター
未定の動作	属性に対する次の未定の動作の指示、例えばリレーショナルデータベースの辞書を変更または未定の動作無し
D B ブロックの個数	この属性に対するリレーショナルデータベースにより使用されたデータベースブロックの個数
列の選択性	この属性に対するリレーショナルコラムにおける値の分配測定
ブロックの選択性	D B ブロックレベルでの分配値の測定
ポインターの記述	ユーザーの便宜のための属性の記述
<p>属性102、“クラスDB名”と同様に、属性102“ポインター属性DB名”はシステム200により発生され、そして、好ましくは、ある長さに制限されると、または単にある特性を持つことに制限されるように、リレーショナルデータベースエンジン240により課せられた名前の制限を満足させることが要求される。“ポインター属性クラスDB名”は、SQLコマンド232、233および261のごとく、この属性に対応するアクセス用リレーショナル構造に対するSQLコマンドを発生させるために、システム200により使用される。好ましい</p> <p>実施では、ユーザー201は、システム200により発生された“ポインター属性DB名”を無効にしてもよいが、リレーショナルデータベースエンジン240により課せられた名前の制約を満足しなければならない。</p> <p>特性102、“表示命令”および“情報の形態”は属性102に対する単なるデフォルト値であってもよく、又、ユーザー201により、特定の表示形態またはサーチが無効にされてもよい。</p> <p>属性102、“DBブロックの個数”、“列の選択”および“ブロックの選択</p>	

”は、ハステム200により演算される。これらの属性102は、リレーショナルデータベースエンジン220を用いてテーブルをサーチするときのシステムの効率を改善するために使用される。

好ましい実施例では、クラスポインター確定クラス305の各オブジェクトは、テーブル3-5の情報を含む。

テーブル3-5

<u>属性</u>	<u>記述</u>
オブジェクトID	このオブジェクトに対するユニークなID
確定クラスの記述	ポインティングとターゲットクラスとの間の関係の記述、“パーツはメーカーにより製作される”、確定クラス記述は“製作される”

クラスポインター確定クラス305の各オブジェクトは、クラスポインタークラス属性304の各オブジェクトをターゲットクラス101にリンクさせる。

サーチ可能な属性：組み込まれた属性

クラスクラス属性303は、組み込みタイプを持つクラスタイプのいかなるオブジェクトに対する各サーチ可能な属性102に対する一つのエンジンを含み、そしてクラスクラス301の各オブジェクトは、クラスクラス属性303のオブジェクトをゼロまたはより多く持つ。クラスクラス属性303のオブジェクトは値を持つ。

好ましい実施例では、クラスクラス属性303の各オブジェクトはテーブル3-6の情報を持つ。

テーブル3-6

<u>属性</u>	<u>記述</u>
オブジェクトID	このオブジェクトに対するユニークなID
属性ラベル	属性のために表示されるべき名前
属性データタイプ	データタイプを指示、この属性は格納される、代数、キャラクタ、データ、整数、通貨、番号またはタイムスタンプのごとく

DB名の属性	ユーザーのリレーショナルデータベースで使用されるべき属性のためのユニークな名前
表示命令	クラス内で他と比較して属性を表示する命令
表示タイプ	データタイプを指示し、この属性が表示、代数、キャラクタ、データ、整数、通貨、番号またはタイムスタンプのごとく
表示長さ	この属性の表示の割り当てのスペース
表示精度	この属性を表示させるためのデジットの個数
ケースのスタイル	テキスト値のみの、常に下側のケース、常に上側のケース、ケース非感覚、ケース感覚に対して
可視列	この属性の表示のための列の個数
関係の意味	ユーザーリレーショナルデータベース内の属性の関係の意味、例えば、一次キー、間隔のキー、記述
情報の形態	いつこの属性が表示されるかの指示、サーチ、サーチ結果、双方、いずれでもないに対して
デフォルト値	ユーザーにより記述されないときの値
値が要求されるか	値が無効かの真相
コラムのインデックスか	この属性に対するコラムがユーザーのリレーショナルデータベース内にインデックスされたかの真相
コラムのインデックスタイプ	リレーショナルデータベース内のインデックスのタイプ
確定値のタイプ	値がある確定値で制約されるかの指示、例えば数えられた確定値または確定値の範囲
外部的な派生か	システムにより発生されたかの真相
正確な表記	数値の値のみに対する、正確な表記で表示され

	るか。
未決定の動作	属性のために未決定の次の動作を指示する、例えばリレーショナルデータベース辞書を替える、または未決定の動作無し
DBブロックの個数	この属性のためにために関係コラムにより使用されたデータベースの個数
列の選択	この属性に対する関係の列における分配値を測定
ブロックの選択	DBブロックレベルでの分配された値の分配の測定
記述の属性	ユーザーの便宜のために属性の記述
<p>属性102、“クラスDB名”、属性102、“属性DB名”は、システム200により発生され、好ましくはリレーショナルデータベースエンジン240により課された名前の制約や、あるキャラクタのみを持つことに制限されるものに満足させることが要求される。“属性クラスDB名”は、SQLコマンド232、233および261のごとく、属性102に対応するリレーショナル構造をアクセスするためのSQLコマンドを発生させるために、システム200により使用</p> <p>される。好ましい実施例では、ユーザー201は、システム200により発生した“属性DB名”を無効にしてもよいが、リレーショナルデータベースエンジン240により課せられた名前の制約を満足しなくてはならない。</p> <p>属性102、“表示命令”、“表示タイプ”、“表示長さ”、“表示精度”、“可視の列”および“情報の形態”は、単に属性102に対するデフォルト値であり、特定の表示形態またはサーチのためにユーザー201により、無効にされてもよい。</p> <p>属性102“DBブロックの個数”、“列の選択”および“ブロックの選択”はシステム200により演算される。これらの属性102は、リレーショナルデータベースエンジン220を使用してテーブルをサーチすることにより、システ</p>	

ムの効率を改善するために使用される。

クラス属性グループ306は、サーチ可能な属性102の各タイプに対して一つのオブジェクトを含み、クラスクラス属性303の各オブジェクトは、ゼロまたはより多くの関係した、クラス101属性グループ306のオブジェクトを持つ。クラス属性グループ306の各オブジェクトは、ひとつのグループ名を持つ。クラスクラス属性303は、一つのグループを単一のフォルダークラス101にリンクする、クラス属性グループ306への関係103を持つ。

サーチ可能な属性データ値

ユーザーのオブジェクトデータベース100では、クラスクラス属性306のサーチ可能な属性102は、整数または浮動小数点のごとき、特定のロジックデータタイプを持ってもよい。メタモデル220では、クラスロジカルデータタイプの決定307は、各ロジカルデータタイプに対して一つのオブジェクトを含み、そして、クラスクラス属性303の各オブジェクトは、クラスロジカルデータタイプの決定307の一つのオブジェクトを持つ。

好ましい実施例はクラスロジカルデータタイプの決定307の各オブジェクトはテーブル3-7の情報を含む。

テーブル3-7

<u>属性</u>	<u>記述</u>
オブジェクトID	このオブジェクトに対するユニークなID
ロジカルデータタイプ名	データタイプに対して表示されるべき名前
DBMSデータタイプ	リレーショナルデータベースエンジンにより決定されたようなデータタイプの名前
DBMSデータタイプ長	リレーショナルデータベースエンジンにより決定されたようなデータタイプの長さ
データ格納の精度	正確なデシマルポイントへのデジットの最大数
データタイプの決定	ユーザーの便宜のためのデータ記述

ユーザーのオブジェクトデータベース100では、クラスクラス属性303のサーチ可能な属性は、特定の範囲にまたは数えられた値の特定の組みに軽減され

た値を持つ。メタモデル220では、クラス数えられた確定値308は、数えられた確定値の各組みに対して一つのオブジェクトを含み、クラスクラス属性303の各オブジェクトは、クラス数えられた確定値308のゼロまたはより多くのオブジェクトに関連する。同様に、クラス確定値の範囲309は、確定値の範囲の各組みに対して一つのオブジェクトを含み、そして、クラスクラス属性303の各オブジェクトは、クラス確定値の範囲309のゼロまたはより多くのオブジェクトに関連する。

好ましい実施では、クラス、数えられた確定値308の各オブジェクトはテーブル3-8の情報を持つ。

テーブル3-8

<u>属性</u>	<u>記述</u>
オブジェクトID	このオブジェクトに対するユニークなID
数えられた値	確定した調査値
確定値の名前	確定した調査値のための名前
カスケードフラグ	確定した調査値が派生クラスに縦につながれた

かの真相

属性102、“カスケードフラグ”は、クラス数えられた確定値308のオブジェクトが、ベースクラス101に対する属性102といつ関連したか否か、数えられた確定値が、ベースクラス101から派生した派生クラス101のオブジェクトに対する同じ属性102と縦つながりになったかを記述する。例えば、図1に示したユーザーのオブジェクトデータベース100の例では、ユーザー201は、クラス101 コンポーネントに対するサーチ可能な属性102、“シリアル番号”を記載してもよく、又、クラス101 コンポーネントに対する数えられた確定値の組みを記載してもよい。しかしながら、このサーチ可能な属性102、“シリアル番号”は、クラスアナログコンポーネントやクラス101 メモリのごとき、ベースクラス101 コンポーネントの派生した各クラスにより、継承されてもよいが、ユーザー201は、これらの派生クラス101が、自身のシリアル番号に対する数えられた値の個別の組を持つように希望してもよい。クラス、

数えられた確定値 308は、クラスクラス 301を有する第1の関係103と、
 クラスクラス属性 303を有する第2の関係103を持つ。

好ましい実施はクラス確定値の範囲 309の各オブジェクトはテーブル3-9
 の情報を持つ。

テーブル 3-9

<u>属性</u>	<u>記述</u>
オブジェクトID	このオブジェクトに対するユニークなID
最小値	最小の確定値
最大値	最大の確定値
デフォルト値	記述なきときはデフォルト値
カスケードフラグ	確定した調査値が派生クラスと縦つなぎになっ たかの真相

属性102、数えられた確定値308に対する“カスケードフラグ”と同様に、属性102、“カスケードフラグ”は、いつ、クラス範囲確定値 309のオブジェクトがベースクラス101に対する属性102と関係したか否か、確定値の範囲が、ベースクラス101から派生した派生クラス101のオブジェクトに対して同じ属性102と縦つなぎになったかを記載する。例えば図1のユーザーのオブジェクトデータベース100の例では、ユーザー201は、クラス101 コンポーネントに対する、サーチ可能な属性102、“シリアル番号”を記載してもよく、又、クラス101 コンポーネントに対する確定値の特定範囲を記載してもよい。このサーチ可能な属性102、“シリアル番号”は、クラスアナログコンポーネントやクラス101 メモリのごとき、ベースクラス101、コンポーネントの各派生クラス101により継承されてもよいが、ユーザー201は、クラス101が自身のシリアル番号に対する値の個別の範囲を持つように希望してもよい。

クラス、確定値の範囲 309は、クラスクラス 301を有する第1の関係103を持ち、そして、クラスクラス属性 303を有する第2の関係103を持つ。

測定ユニット

ユーザーのオブジェクトデータベース100では、クラスクラス属性303のサーチ可能な属性102は、測定値の明示されたユニット内で表された値を持つようにしてもよい。メタモデル220では、クラス、測定ユニット310は各測定ユニットに対して一つのオブジェクトを含み、そして、クラスクラス属性303の各オブジェクトは、クラス測定ユニット310の少なくとも一つと(ディスプレイに対して)関係し、又、クラス測定ユニット310の少なくとも一つのオブジェクトと(格納のために)関係している。

好ましい実施では、クラス、測定値のユニット310の各オブジェクトはテーブル3-10の情報を含む。

テーブル3-10

属性

オブジェクトID

UOM名

UOM表示シンボル

UOM記述

記述

このオブジェクトのためのユニークなID

測定ユニットの名前

測定ユニットのために表示されるべきシンボル、例えば“mg”

ユーザーの便宜のための、ソクテイユニットの記述

ユーザーのオブジェクトデータベース100では、第1の測定ユニットは第2の測定ユニットにに変換される。例えば、1000をかけてボルト→ミリボルトに変換する。メタモデル220では、クラスユニット変換公式311は、測定ユニットを変換するための各公式に対する一つのオブジェクトを含み、クラスユニット変換公式311の各オブジェクトは、クラス測定ユニット310のゼロもしくははより多くのオブジェクトと(ソースユニットのために)関係しており、そして又、クラス測定ユニット310のゼロもしくははより多くのオブジェクトと(ターゲットユニットのために)関係している。

カスタム機能

カスタム機能は、システム200内のオブジェクトを流動的に変形させるための技術を提供し、メタモデル220内にオブジェクトを含む。これらのカスタム

機能は、挿入イベント、変形イベントおよび削除イベントによりトリガーされる。ユーザー201は、カスタム機能が呼び出されたとき、システム200により、実際の変化を記述する。

メタモデル220は、計算、クラスクラス属性機能操作312、クラスクラス機能操作、クラスカスタム機能決定314、外部アクセス機能操作315およびクラス公式316のためのオブジェクトを具体化するいくつかの付加的クラス101を含む。

クラスユニット変換公式311はソース測定ユニット311に対するクラス測定ユニット310を有する第1の関係103と、ターゲット測定ユニットのため

のクラス測定ユニット310を有する第2の関係103と、変換公式のためのクラス公式316を有する関係103を持つ。

クラス公式316のオブジェクトは、代数式を含み、そして公式名、加数、倍率器、加算／掛算のルール(加算、掛算、加算後に掛算または掛算後に加算)および公式の記述を含む。

クラスクラス属性機能操作312およびクラスクラス機能操作313のオブジェクトは、リレーショナルデータベース250のための操作に加えて、もしくはその代わりに、実行されるべき機能についての情報を含み、又、操作の順序(例えば操作システム上の呼び出しの言葉で定義される)および、カスタム機能が加算するか置き換えるDBMS操作を含む。

クラスカスタム機能決定314は、カスタム機能に関する情報を含み、又、機能名、バージョン番号、機能記述、カスタム機能に関係するクラス101のオブジェクトが創造されたときにカスタム機能の実行を強行するフラグ、派生クラス101のオブジェクトが創造されたときにベースクラス101のためのカスタム機能が無効にするためのフラグを持つ。カスタム機能は、クラス101またはクラス101内のオブジェクトのために、クラス番号機能を加算または無効にし、又、クラスメンバー機能の通常後に操作の前か後にクラスメンバー機能が無効にするために記述されてもよい。

クラス外部アクセス機能操作315のオブジェクトは、ファイルシステムオブ

ジェクト、外部データベース、オペレーションシステム上の呼び出しによりアクセスされる別のオブジェクトのごとく、システム200とは外部のオブジェクトをアクセスするための機能に関する情報を含む。

クラス情報表示

好ましい実施例では、クラスクラス形態320は、ユーザーのオブジェクトデータベース100内のクラス101の表示形態の各記述のための一つのオブジェクトを含む。クラスクラス形態320の各オブジェクトは、クラスサーチペイン(窓ガラス)形態321のゼロまたはより多くのオブジェクトに関係し、クラス101は、クラス101上のサーチに対する各表示形態のための一つのオブジェクト

を含み、又、クラス結果ペイン形態322のゼロまたはより多くのオブジェクトに関係し、クラス101は、クラス101上のサーチの結果に対する各表示形態のための一つのオブジェクトを含む。

好ましい実施では、クラスサーチペイン形態321の各オブジェクトおよびクラス結果ペイン形態322の各オブジェクトはテーブル3-11の情報を含む。

テーブル3-11

<u>属性</u>	<u>記述</u>
オブジェクトID	このオブジェクトに対するユニークなID
表示タイプ	そのフィールドの表示に対するデータタイプ
表示幅	そのフィールドに対する最大表示幅
表示UOM	表示のための測定値
列の命令	表示している列に対する相対命令
コラム命令	表示しているコラムに対する相対命令
カスタムラベル	フィールドを表示するためのラベル
<p>クラス<u>サーチペイン形態321</u>の各オブジェクトおよびクラス<u>結果ペイン形態322</u>の各オブジェクトもまた、表示されるべきクラスおよびクラス属性を決定するために、クラス<u>クラス301</u>のゼロまたはより多くのオブジェクトと関係し、そして、クラス<u>クラス属性303</u>のゼロまたはより多くのオブジェクトと関係する。</p>	

アクセスコントロール

図4はユーザーのオブジェクトデータベースに対するアクセスコントロールをモデリングするための1組みのクラスのデータモデルダイヤグラムを示す。

好ましい実施例では、図4に示された組みのクラスは、システム200のために予め定義される。しかしながら、メタモデル220と違って、これらのクラス101は、アクセスコントロールをモデリングする際の都合のために、提供され、そして、ユーザーのオブジェクトデータベース100をモデリングし、翻訳し、

またはサーチするために必ずしも必要としない。

クラスルート401は、すべてのクラス101に対するベースクラスである。好ましい実施例では、クラスルート401は、システム200内のすべてのオブジェクトに属する、ほんの少しのサーチ可能な属性を持ち、最後の創造のオブジェクトまたは最後に変形されたオブジェクトの署名、タイムスタンプ、オブジェクトが特定のユーザー201かを指示するためのプライベートフラグ、および、オブジェクトがクラステンプレートのメンバーならばテンプレート識別子のごときである。クラスルート401は、クラスルート401を有する関係がすべてのクラス101により継承されるように提供される。

クラスユーザー402のオブジェクトは、システム200の個々のユーザーを示し、そしてユーザーについての情報、ユーザー名、ユーザーのパスワード(セキュリティのために極力エンコードされる)、“スーパーユーザー”の地位、郵便アドレス、電話番号および、ユーザーの代金をだれが負担かのごとき計算用情報、を含む。

クラスユーザーグループ403のオブジェクトは、システム200のユーザーのグループを示し、そのすべては同様なアクセスコントロール権利を持つ。親子の体系からのユーザーグループ(より詳しくは指示された周期的グラフ)は、クラス101の親子の関係を好む。クラスユーザーグループ403は、多くから多くの関係と、自身がクロスリンクを使用するクラスユーザーグループからユーザーグループ404を持ち、それはユーザーグループの親子の体系を示す。

クラスユーザー402およびクラスユーザーグループ403は、クロスリンククラスユーザーからユーザーグループリンク405を用いる、多くから多くの関係を持ち、これは、ゼロまたはより以上のユーザーグループにおける、ゼロまたはより多くのユーザーのメンバーシップを示す。

アクセスコントロールは、二つの特徴が与えられる。第1の特徴はクラス属性または個々の属性の所有であり、その所有は、ユーザーグループまたはユーザーによる。第2の特徴は、クラスまたは個々のオブジェクトに対するエディットの許可であり、そのエディットの許可はユーザーグループまたはユーザーによる。

所有権は、クロスリンククラスユーザーグループからクラス所有権406、クロスリンククラスユーザーグループからクラス財産所有権407、クロスリンククラスユーザーグループからオブジェクト所有権408、クロスリンククラスユーザーからクラス財産所有権409およびクロスリンククラスユーザーからオブジェクト所有権410により示される。所有権は、この結果、多くから多くの関係であり、この関係は、クラスクラス財産およびオブジェクトをそれらのユーザーグループおよびユーザーの所有権にリンクする。

エディットの許可は、クラスユーザーグループからクラスオブジェクトエディット許可411、クラスユーザーグループからオブジェクトエディット許可412、クラスユーザーからクラスオブジェクトエディット許可413およびクラスユーザーからオブジェクトエディット許可414により示される。エディット許可はまた、クラス財産およびオブジェクトをユーザーグループおよび許可を有するユーザーにリンクする多くから多くの関係であり、このような各関係に対して特定のエディット許可はエディット許可オブジェクト内にリストされることを除く。

グラフィカル情報表示

図5は、ユーザーのオブジェクトデータベースのための予め定義されたグラフィカル情報の表示に対する1組みのクラスのデータモデルダイアグラムを示す。

好ましい実施例では、図5により示したクラスの組みは、システム200のために予め定義される。しかしながら、メタモデル220とは似ず、これらのクラ

ス101は、予め定義されたグラフィカル情報をモデリングする際に便宜上、与えられ、ユーザーのオブジェクトデータベース100をモデリングし、翻訳し、又はサーチするためには必ずしも必要としない。

クラスグラフィックス501のオブジェクトは、予め定義されたグラフィックスのオブジェクトについての情報を含む。クラスグラフィックス501は、クロスリンククラスオブジェクトリンクへのグラフィックス502により、クラスルート401に関係し、多くから多くの関係を定義し、その関係では、いかなるオブジェクトも分担してあらゆるグラフィックスオブジェクトに付随され、そして、いかなるグラフィックスオブジェクトもいかなるオブジェクトに付随されてもよい。

い。クラスグラフィックス501は、多くからひとつの関係により、クラスルート401に関係し、その関係では、グラフィックスオブジェクトはいかなるオブジェクトと非分担であってもよい。

クラス別名変数503は操作システムを使用できるファイルに関する情報を含む。クラスグラフィックス501は、多くからひとつの関係により、クラス別名変数503に関係し、グラフィックスオブジェクトを実行するために、ファイルに対する各々のひとつはシステムを操作することにより使用される。同様に、クラスバイナリー ラージ オブジェクト504のオブジェクトは、操作システムにより実行されるバイナリーラージオブジェクトを含み、ゼロまたはより多くのグラフィックスオブジェクトに関係する。

クラスグラフィックス501もまた、多くからひとつの関係103により、クラスユーザー402に関係し、その関係では、グラフィックスオブジェクトはユーザーにより、“チェックアウト”または“チェックイン”(ライブラリの本のように)されてもよい。このことは、グラフィックスオブジェクトが再び他のユーザー201に利用可能となったとき、ユーザー201がグラフィックスオブジェクトをチェックアウトし、自動的にそれをアップデートし、そしてグラフィックス オブジェクト バックをチェックすることを可能にする。

クラスローカル505のオブジェクトは、国名のごとき論理位置を含む。ゼロ

またはより多くの位置は、クラス測定ユニット 310のオブジェクトにより示された通貨や、クラスタイムゾーン 506のオブジェクトに関係してもよい。クラスローカル 505のオブジェクトは、言語識別子、通貨シンボルおよびその位置、番号のフォーマット(例えばデジットセパレータ、データフォーマット、時間フォーマット、基数のポイント)、“誤り”、“真実”の語を含む。ゼロまたはより多くのユーザーは、ひとつのローカルに関係してもよい。

クラス情報テキスト 507のオブジェクトは、いずれかのオブジェクトについてのテキスト情報を含む。クラス情報テキスト 507は、多くからひとつの関係103により、クラスルート 401に関係し、そこでテキスト情報はいずれかのオブジェクトに所属されてもよい。クラス情報テキストタイプ 508は、テキス

ト情報のための描写タイプ(例えば“eメール”または“メモ”)を含む。クラス情報テキストタイプ 508は、多くからひとつの関係103により、クラス情報テキスト 507と関係する。

システム200は又、各オブジェクトに所属してもよい“ノート”を提供する。オブジェクトと関連しなくてはならないテキスト情報は、短く、例えば数ラインのテキストであり、ユニットを有するユーザー201は単に、短いコメントまたはリマインダーのごとき“ノート”をオブジェクトに加算する。この特徴は、ユーザー201が、オブジェクトに関係したノートのためのサーチ可能な属性ユーザー201を創造するのに役立つ。

クラス活動 509のオブジェクトは、“署名”のごときユーザーのための、活動クラスを含む。クラスユーザー 402は、クロスリンククラスユーザーリンクへの活動510を使用するクラス活動 509を有する多くから多くの関係を持つ。同様に、クラスユーザーグループ 403は、クロスリンククラスユーザーグループへの活動511を使用するクラス活動 509を有する多くから多くの関係を持つ。

好ましい実施例では、システム200は、いくつかまたはすべてのアタッチメントが、同じ操作システムの制御下では他のアプリケーションによりアクセスできない(システム200内から起こされた他のアプリケーションを除く)、センタ

一位置に格納される。

データモデルの構築およびエディット

図6は、ユーザーのオブジェクトデータベースのデータモデルを構築しエディットするための工程のフローダイアグラムを示す。フローポイント600では、ユーザー201は、ユーザーデータベースモデル230を構築またはエディットすることを希望する。

好ましい実施例では、システム200は、ユーザー201により呼び出されるべき“処理開始”コマンドおよび“処理終了”コマンドを使用して自動処理のための容易性を提供する。ユーザー201が“処理開始”コマンドを呼び出したとき、ユーザーデータベースモデル230へのすべての変化またはリレーショナル

データベース250へのすべての変化が集められ、自動操作に委ねられる。この特徴は、ユーザー201がユーザーデータベースモデル230およびリレーショナルデータベース250を部分変化に対してプロテクトするのに役立つ。

このフローダイアグラムは、システム200がユーザーのコマンドを試験する際、ユーザー201が命令を述べる実際の実行の際の命令に関して述べられており、このフローダイアグラムに指示された命令に必ずしも従う必要はない。

ステップ610では、システムは、ユーザー201がクラス101を構築、エディットまたは削除するかの選択をしたか決定する。もし、そうならば、システム200は、ステップ612に進む。そうでないならば、システム200はステップ620に進む。

ステップ612では、ユーザー201は、クラス101の構築、エディットまたは削除を選択する。もし、ユーザー201が新しいクラス101の構築を選択したとき、システム200は、ステップ614に進む。もしユーザー201が存在するクラス101のエディットを選択したならば、システム200はステップ616に進む。もしユーザー201が存在するクラス101の削除を選択したときは、システム200はステップ618に進む。

ステップ614では、システム300は、メタモデル220内にクラスクラス301の新しいオブジェクトを創造する。システム200は、ユーザー201に

対して、今創造されたクラス クラス 301 の新しいオブジェクトの属性 102 に対して値を選択する能力を与える。“クラスDB名”のごときこれらのいくつかの属性のために、システム200は、デフォルト値を発生させるが他のものに対しては、ユーザー201が供給した値を要求する。システム200はその後ステップ610に戻る。

ステップ616では、ユーザー201は、メタモデル220内のクラス クラス 301 のオブジェクトを選択し、そしてそのオブジェクトをエディットするために進む。システム200は、エディット用コマンドを受け取り、そしてオブジェクトの属性 102 に対する値を変更する。

ステップ618では、ユーザー201は、メタモデル220内のクラス クラス 301 のオブジェクトを選択する。システム200は、メタモデル220からオブジェクトを移動し、そして、いかなる削除ルールも実行する。クラス クラス 301 の前記オブジェクトに関するクラス クラス属性 303 のオブジェクトを削除するようにして。システムはその後、ステップ610に戻る。

ステップ620では、システムは、ユーザー201がクラス属性 102 を構築、エディットまたは削除するかの選択したかを決定する。もし、そうならば、システム200は、ステップ622に進む。そうでないならば、システム200はステップ630に進む。

ステップ622では、ユーザー201は、クラス属性 102 の構築、エディットまたは削除を選択する。もしユーザー201がクラス属性 102 の構築を選択したならば、システム200はステップ624に進む。もしユーザー201がクラス属性 102 のエディットを選択したならば、システム200はステップ626に進む。もしユーザー201がクラス属性 102 の削除を選択したならば、システム200はステップ628に進む。(当然、クラス属性 102 はクラス 101 に関係しているので、これらのどの動作もクラス 101 をエディットする影響を持つ)

ステップ624では、システム200は、メタモデル220内のクラス クラス属性 303 の新しいオブジェクトを創造する。システム200は、ユーザー20

1 に対して、今創造されたクラス クラス属性 3 0 3 の新しいオブジェクトの属性 1 0 2 に対して値を選択する能力を与える。“クラスDB名”のごときこれらのいくつかの属性のために、システム 2 0 0 は、デフォルト値を発生させるが他のものに対しては、ユーザー 2 0 1 が供給した値を要求する。システム 2 0 0 はその後ステップ 6 1 0 に戻る。

ステップ 6 2 6 では、ユーザー 2 0 1 は、メタモデル 2 2 0 内のクラス クラス属性 3 0 3 のオブジェクトを選択し、そしてそのオブジェクトをエディットするために進む。システム 2 0 0 は、エディット用コマンドを受け取り、そしてオブジェクトの属性 1 0 2 に対する値を変更する。システムはその後ステップ 6 1 0 に戻る。

ステップ 6 2 8 では、ユーザー 2 0 1 は、メタモデル 2 2 0 内のクラス クラス属性 3 0 3 のオブジェクトを選択し、そしてそのオブジェクトを削除するために進む。システム 2 0 0 は、メタモデル 2 2 0 からのオブジェクトを移動し、そしていかなる削除ルールをも実行する。システムはその後ステップ 6 1 0 に戻る。

ステップ 6 3 0 では、システムは、ユーザー 2 0 1 がクラス 1 0 1 間の関係 1 0 3 を構築、エディットまたは削除するかの選択したか決定する。もし、そうならば、システム 2 0 0 は、ステップ 6 3 2 に進む。そうでないならば、システム 2 0 0 はステップ 6 4 0 に進む。

ステップ 6 3 2 では、ユーザー 2 0 1 は、クラス 1 0 1 間の関係 1 0 3 の構築、エディットまたは削除を選択する。もしユーザー 2 0 1 がクラス 1 0 1 間の新しい関係 1 0 3 の構築を選択したならば、システム 2 0 0 はステップ 6 3 4 に進む。もしユーザー 2 0 1 がクラス 1 0 1 間の存在する関係 1 0 3 のエディットを選択したならば、システム 2 0 0 はステップ 6 3 6 に進む。もしユーザー 2 0 1 がクラス 1 0 1 間の存在する関係 1 0 3 の削除を選択したならば、システム 2 0 0 はステップ 6 3 8 に進む。(当然、クラス 1 0 1 間の関係 1 0 3 は、ポインタークラス属性を含み、クラス 1 0 1 に関係しているので、これらのどの動作もクラス 1 0 1 をエディットする影響を持つ)

ステップ 6 3 4 では、システム 2 0 0 は、メタモデル 2 2 0 内のクラス ポイン

タークラス属性 304の新しいオブジェクトを創造する。システム200は、ユーザー201に対して、今創造されたクラスポインタークラス属性 303の新しいオブジェクトの属性102に対して値を選択する能力を与える。“クラスポインター属性DB名”のごときこれらのいくつかの属性のために、システム200は、デフォルト値を発生させるが他のものに対しては、ユーザー201が供給した値を要求する。システム200はその後ステップ610に戻る。

ステップ636では、ユーザー201は、メタモデル220内のクラスポインタークラス属性 304のオブジェクトを選択し、そしてそのオブジェクトをエディットするために進む。システム200は、エディット用コマンドを受け取り、そしてオブジェクトの属性102に対する値を変更する。システムはその後ステップ

610に戻る。

ステップ638では、ユーザー201は、メタモデル220内のクラスポインタークラス属性 304のオブジェクトを選択し、そしてそのオブジェクトを削除するために進む。システム200は、メタモデル220からのオブジェクトを移動し、そしていかなる削除ルールをも実行する。システムはその後ステップ610に戻る。

ステップ640では、ユーザー201がメタモデル220内のいずれもの別のオブジェクトを構築、エディットまたは削除することを望まないとき、この工程は完了する。

、 工程のコンパイルおよび翻訳

図7は適したオブジェクトおよびリレーショナルデータベース構造間のコンパイルおよび翻訳のための工程のフローダイアグラムを示す。

フローポイント700では、ユーザー201がユーザーのオブジェクトデータベース100を対応するリレーショナルデータベース203へ翻訳することを希望する。

ステップ710では、完成したエディット用のユーザーモデル230を持つユーザー201は、翻訳工程をトリガーする。

ステップ720では、システム200は、ユーザーのオブジェクトデータベース100内のクラス101を選択する。

ステップ730では、システム200は、選択されたクラス101に対応するリレーショナルデータベース230内にテーブルを創造する。

ステップ740では、システム200は、テーブル内に1組みの特定のコラムを創造する。これらの特定の Colum は、各オブジェクトに対するユニークな識別子(“UID”)のためのひとつの Colum、各オブジェクトに対する“クラス識別子”に対するひとつの Colum、および各オブジェクトに対する“オブジェクトタイプ”に対するひとつの Colum を含む。

ひとつのオブジェクトが対応するリレーショナルデータベース203に取り込まれたとき、システム200は、オブジェクトのクラス101に対応するテーブル

内と、オブジェクトのクラス101に対するベースクラスに対応する各テーブル内とに列を創造する。これにより、各オブジェクトは、オブジェクトのクラス101に対するベースクラス101である各クラス101に対応する各テーブル内のひとつの列に対応する。

各テーブル内の第1の Colum は、各オブジェクトに対するUID用のものである。これにより、オブジェクトに対して列が創造された各テーブル内では、第1の Colum がオブジェクトに対するUIDに対応する。多くのリレーショナルデータベースエンジンは、第1 Colum に対してインデックスされたサーチを行うようになされているので、この Colum は、好ましくは第1のほとんど左の Colum に選択される。

各テーブル内の別の Colum は、各オブジェクトに対する“クラス識別子”に対するものである。ひとつのオブジェクトは、クラス101に対するテーブル内と、クラス101に対するベースクラス101に対する各テーブル内との列に対応するので、各オブジェクトがどのクラス101であるかの識別子を、前記テーブルに対する列が格納される各テーブル内に記録するのが効率的である。

各テーブル内の別の Colum は、各オブジェクトに対する“オブジェクトタイプ”に対するテーブルである。サーチ可能な属性102、クラスクラス301内の

“クラスタイプ”のように、“オブジェクトタイプ”のコラムは、この特定のオブジェクト(全体のクラスに対して提案されたように)がメタモデルの一部か、あるいはユーザーにより創造されたオブジェクトであるかを指示する。

ステップ750では、システム200は、クラス101に対するサーチ可能な属性102に対する値のためのテーブル内にコラムを創造する。もしサーチ可能な属性102が組み込みのデータタイプを持つならば、コラムは組み込みのデータタイプに対する実際の値を保持する。そうでなく、もし、サーチ可能な属性102がユーザー定義のデータタイプを持つならば、コラムは、前記ユーザー定義のデータタイプに対する値に対応するオブジェクトのためのUIDを保持する。

ステップ760では、システム200は、クラス101と関係するクラス101との間の各関係に対するテーブル内にコラムを創造する。この関係103に対

するコラム内では、システム200は、このテーブルに対するクラス101内のオブジェクトに関係するクラス101内のオブジェクトのUIDを置く。もし、その関係103が多くからひとつであれば、そのシステムは、同一のオブジェクトに対するテーブル内に多数の記録を形成し、その各記録は、関係クラス101内の子となったオブジェクトのUIDを持つ。

フローポイント770では、コンパイルおよび翻訳工程が完了する。

ターゲットリレーショナルデータベースのサンプル

図8は、図1のユーザーのオブジェクトデータベースの例に対応するユーザーリレーショナルデータベースのデータモデルダイアグラムを示す。

図7に関して述べたように、ユーザーのオブジェクトデータモデルダイアグラム100に対応するユーザーリレーショナルデータベースは、1組みの関係テーブル900を含み、各々はクラス101に対応している。従って図1の例では、システム200は、テーブル800 コンポーネント、テーブル800 アナログコンポーネント、テーブル800 メモリ、テーブル800 ダイナミックメモリ、テーブル800 スタティックメモリ、テーブル800 メーカー、テーブル800 国内メーカー、およびテーブル800 外国メーカーを創造する。

このような各テーブル800は、1組みのコラム810と1組みの列820を

含む。

システム200は、各テーブル800に対して、オブジェクトのU I Dに対するコラム810“オブジェクトI D”を形成し、そして派生クラス101に対する各テーブル800に加算されたオブジェクトも又、対応するU I Dを、ベースクラス101に対するテーブル800に加算されることを確実にする。例えば、テーブル800 スタティックメモリに加算される各列820は、テーブル800 メモリに加算される列820に対応し、テーブル800 コンポーネントに加算される列820に対応し、そして複数のU I Dは、これらの3つの列820に対するコラム810“オブジェクトI D”内において同一である。

システム200は、又、テーブル800 コンポーネントに対して、コンポーネントの名前に対するコラム810“名前”を、テーブル800 メモリに対して、

コンポーネントのサイズに対するコラム810“サイズ”を、テーブル800 メーカーに対して、メーカー名に対するコラム810“名前”およびその都市に対するコラム810“年”を、そしてテーブル800 外国メーカーに対しては、外国メーカーの国に対するコラム810“国”を創造する。

システム200は、又、テーブル800 コンポーネントに対して、テーブル800 メーカー内の列820のU I Dに等しい、メーカーのU I Dに対するコラム810“メーカー”を創造し、これにより、クラス101 コンポーネントおよびクラス101 メーカーのオブジェクトとの間の関係103を確立する。

縦方向のサーチおよびサーチ翻訳

図9は、オブジェクトデータベースの縦方向のサーチおよび適応されたオブジェクトとリレーショナルデータベースとの間のサーチ翻訳に対する工程のフローダイアグラムを示す。

フローポイント900では、ユーザー201は、ユーザーのオブジェクトデータベース100の縦のサーチを実行することを希望する。

縦方向のサーチの記述

ステップS910では、ユーザー201は、縦方向のサーチ要求を記述する、つまり、クラスが縦につながれ、そして各クラス内で属性がサーチされる。ユー

ザーインタフェイス210は、ユーザー201から情報を受信する。このステップ910を実行するために、システム200は、ステップ911から916を含めて実行する。

ステップ911では、ユーザーインタフェイス210は、ユーザーデータベースモデル230内のクラス101のリストをメタモデル220から得て、そして、クラス101のリストを選択のためにユーザー201に与える。

図1の例のユーザーのオブジェクトデータベース100を用いた縦方向のサーチの例では、システム200は、ユーザーデータベースモデル内で利用できるクラス101のリストを与える。

ステップ912では、ユーザー201は、クラス101を選択する。ユーザーインタフェイス210は、ユーザーデータベースモデル230内の関連するクラ

ス101のリスト(つまり、関係103により選択されたクラス101に結合したクラス101)をメタモデル220およびそれらの派生クラス101から得る。ユーザーインタフェイス210は、関連したクラス101のリストと、それらの派生クラス101を選択のためにユーザー201に与える。

縦のサーチ例では、ユーザーはクラス101 スタティックメモリを選択できる。クラス101 スタティックメモリは、その親クラス101 メモリの関係103のすべてと、クラス101 メーカーとの関係を持つ全体の親クラス101 コンポーネントのすべての関係とを継承する。

ステップ913では、ユーザー201は、関連したクラス101または関連したクラス101から派生したクラス101を選択してもよい。ユーザー201が関連したクラスまたはそれらから派生したクラス101を選択する限り、システム200は、ステップ912を継続する。ユーザー201がいずれか別のクラス101またはそれらから派生したクラス101を選択するようになったとき、システムはステップ914を継続する。

この縦方向のサーチ例では、ユーザー201は、クラス101 外国メーカーを選択することができ、それは、関連したクラス101 メーカーから派生したクラス101である。

ステップ914では、ユーザー201により選択された各クラス101に対して、ユーザーインターフェイス210は、メタモデル220からのユーザーデータベースモデル230内でサーチ可能な属性のリスト(およびそれらの値におけるいずれかの制限)を得て、そして、選択のために、サーチ可能な属性のリストをユーザー201に与える。

この縦方向のサーチ例では、システム200は、クラス101 スタティックメモリに対するサーチ可能な属性102“名前”および“サイズ”を提供し、そして、クラス外国メーカーに対するサーチ可能な属性“102”および“国”を与える。

ステップ915では、ユーザー201は、選択されたクラス101の少なくともひとつのサーチ可能な属性を選択し、それらの値に制限を記載し、そして選択

されたクラス101の選択可能な属性102の選択を継続してもよい。ユーザー201がサーチ可能な属性102の選択を継続する限り、システム200はステップ914を継続する。ユーザー201がいずれか別のサーチ可能な属性102を選択するようになったときは、システムはステップ916を継続する。

この縦方向のサーチ例では、ユーザー201は、提供されるべきクラス101 スタティックメモリに対するサーチ可能な属性102“名前”を選択でき、そして、サーチされるべきクラス101 外国メーカーに対するサーチ可能な属性“国”を選択し、そして、後者はテキスト記号列“フランス”と等しくなければならない。

好ましい実施例では、ステップ914および915は、ステップ912およびステップ913と並行に進んでもよい。ユーザー201がクラス101を選択したとき、システム200は、選択のために、前記クラス101内のサーチ可能な属性を提供し、そして、又、選択のために関連したクラス101(および派生したクラス101)も提供する。敏速に応答する代わりに、ユーザー201は、その後、関連したクラス101(または派生クラス101)かサーチ可能な属性102を選択するために、指示デバイスを用い、あるいは、縦のサーチ(ステップ916における)をトリガーするためのコマンドを選択する。

ステップ916では、ユーザー201は、縦方向のサーチが十分に記載され、サーチをトリガーするように指示する。システム200は、ユーザー201が縦方向のサーチの記述を記載し、過去に記録された縦方向のサーチの記載を呼び出し、新しい縦方向のサーチの記載をエディットし、そして、現在のリレーショナルデータベース250に対する過去に記録された縦方向のサーチを適用できるように技術を与える。

この縦方向のサーチ例では、ユーザー201は、フランスのメーカーにより作成されたすべてのスタティックメモリの名前に対する縦方向のサーチを要求する。

質問モデルの構築

ステップ920では、システム200は、縦方向のサーチ要求を理解し、そして質問モデル260を構築する。このステップ920を実行するために、システム

200は、ステップ921から922を含めて実行する。

ステップ921では、システム200は、サーチされるべきユーザー201により選択された初期のクラス101のための形態オブジェクト991を構築し、そして、サーチされるべきサーチ可能な属性102の第1のリストと、質問結果251内に表示されるべきサーチ可能な第2のリストと、ユーザー201により選択された初期のクラス101でスタートするクラス101の第3のリストを含み、ユーザー201により選択された各々の関係するクラス101を継続する。

ステップ922では、システム200は、サーチされるべきユーザー201により選択された各々の付随的なクラス101のための形態オブジェクト991を構築し、そしてこのような付随的形態オブジェクト991を、サーチされるべきユーザー201により選択された初期のクラス101に対する形態オブジェクト991にリンクする。

SQL質問を構築する

ステップ930でシステム200は、質問モデル260に回答してSQL質問261を構築する。このステップ930を実行するために、システムはステップ931ないし936を含めて実行する。

ステップ931でシステム200は、形態オブジェクト991を、質問モデルオブジェクト992のリスト内へ再リンクする。各質問モデルオブジェクト992は、縦方向のサーチの単一のサーチフィールドまたは単一の結果フィールドを記述する。

ステップ932でシステム200は、質問モデル260に対する別の分析テーブルを実行する。

このステップ932を実行するために、システム200は、質問モデルオブジェクト992のリストを試し、そして、質問モデル206のSQLコマンド261への翻訳が、外の点で、いずれかのトラブルがそれ自身と結合させたかを決定する。つまり、同一テーブルのひとつまたはより多くのコラムが質問モデル260内に現れたかを決定する。もしそうならば、システム200は、各別名に対する別名の記録993を最初に作った後に、前記テーブル例の各々に対するオラクル

“別名”に対するユニークな名前を発生し、そして、別名記録993を質問モデルオブジェクト992に装着する。この別名記録993は、別名が記述されなければならないことを示し、又、発生されるべきSQLコマンド内の別名を使用することに関係するテーブルを示す。

ステップ933でシステム200は、質問モデル260に対する結合した分析を実行する。

このステップ933を実行するために、システム200は、質問モデルオブジェクト992のリストを試し、そして、質問モデル206のSQLコマンド261への翻訳が、外の点で、いずれかの結合状態の写しが記述されたかを決定する。つまり、もし第1および第2のテーブルが同一の結合条件を用いて2倍になったかが判定される。もしそうならば、システム200は、適用されるべき結合状態のただひとつを選択し、このような各結合状態に対する結合記録994を創造し、そして、結合記録994を質問モデルオブジェクト992に装着させる。結合記録994は、単一の結合状態が記述されることと、発生されるべきSQLコマンド内にて結合状態が省略されるべきことを示す。

ステップ934でシステム200は、質問モデル260に対するサーチ状況分析を実行する。

このステップ934を実行するために、システム200は、テーブル9-1に示されたサーチ状況の各々に対する質問モデルオブジェクトを試し、各サーチ状況に対する状況記録を創造し、そして、状況記録995を質問モデルオブジェクト992に装着する。

テーブル9-1

・“WHERE”節を用いたSQLコマンド内に示された数値番号とサーチ可能な属性との比較および、前記数値番号とリレーショナルデータベース250内のコラムの論理比較。

・“WHERE”節を用いたSQLコマンド内に示されたテキスト記号列とサーチ可能な属性との比較および、前記テキスト記号列とリレーショナルデータベース250内のコラムの論理比較。もし比較されるべきテキスト記号列が“悪い

カード”(例えば一つまたはより多くの独断的なキャラクターのマッチングを示すための特別なキャラクター)を含むならば、その比較は、オラクル“LIKE”の一覧表を用いたSQLコマンド内で示される。もし比較されるべきテキスト記号列が非感知のケースであるならば、その比較は、ケースを上部ケースとするためのオラクル声明を使用したSQLコマンド内に示される。他の記号列は、変換テーブルを用いて適したオラクル声明に変換される。

・メタモデル220内に記録されたように、ソースおよび測定ユニットのターゲットに対する適した変換機能を用いたSQLコマンド内に示された、異なった測定ユニットを用いたサーチ可能な比較。

・サーチ可能な属性または、“AND”、“OR”および“NOT”のごとき2値論理オペレータを用いたSQLコマンド内で示された比較における他の論理オペレーションの複数の比較。

・異なった数値またはテキスト記号列の値と同じサーチ可能な属性との多数の比較を用いたSQLコマンドに示された、数字のグループまたはテキスト記号列の値または数値の範囲とサーチ可能な属性との比較。

好ましい実施例では、縦方向のサーチが測定ユニットを持つサーチ可能な属性 102 と関係するとき、システム 200 は、サーチ可能な属性 102 と、比較される値とに対する測定ユニットを試し、そして、もし、それらが同一でないとき、異なる測定ユニットを説明するために、SQL コマンド 261 を調節する。

この調整を行うために、システム 200 は、サーチ可能な属性 102 のために クラスクラス属性 303 のオブジェクトを試し、そして、クラスクラス属性 303 のオブジェクトに関係(格納測定ユニットとして)した、クラスクラス測定ユニット 310 のためにメタモデル 200 をサーチする。システム 200 は、その後、質問モデル 260 を試し、縦方向のサーチのために測定ユニットを決定し、そして、測定ユニットと関係した クラス測定ユニット 310 のオブジェクトに対するメタモデル 200 をサーチする。システム 200 は、その後、第 1 の測定ユニットを第 2 の測定ユニットに変換するクラスユニット変換公式 311 のオブジェクトに対するメタモデル 200 をサーチし、そして、前記ユニット変換公式への呼

び出しを SQL コマナド 261 内へ挿入する。

ステップ 935 でシステム 200 は、質問モデル 260 に対する最高活用した分析を実行する。

このステップ 935 を実行するために、システム 200 は、テーブル 9-2 に示された最高に活用できる状態の各々に対する質問モデルを実行し、そこに示された最高活用の技術に基づく SQL コマンド 261 を発生させるために質問モデル 260 を変形する。

テーブル 9-2

・列の選択。“列選択性”のための値は、次の公式に基づき、各々のサーチ可能な属性により、演算される。

列選択性 = 明確な値 K # / 0 値を持たないオブジェクトの #

列選択性は、%として標準化するために 100 倍され、サーチ可能な属性をモデリングするコラムがインデックスされるべきかを決定するために使用される。もし、コラムの列選択性が 70 % より大きいならば、コラムはほとんどの場合に

インデックスされ、もし、コラムの列選択性が30%程度ならば、そのコラムは決してインデックスされない。

・並び替え／混合の回避。WHERE節内のインデックスされた多数のコラムを使用することは、並び替え／混合の操作を実行するためにオラクルRDBMSを生じさせる。並び替え／混合の操作は実質的な時間をとるので、システム200は、このような構造を、ただ一つのインデックスされたコラムを使用するWHERE節と置き換える。

オラクルRDBMSにより処理されるべきインデックスされたコラムをインデックス無しとするために、サーチされるべき値は、列コラム値、テーブルコラムから演算された値、テーブルコラム+0(数値に対する)、またはテーブルコラム(テキスト記号列値)から変化される。

そうではなく、コラムがインデックスされるなら、システムは、前記コラムを

インデックスされたものとして扱うために、オラクルRDBMSを要求するためのSQLコマンドを書くことを試みる。コラム値に機能または算術演算の操作が適用されたとき、オラクルRDBMSは、コラムをインデックスされたものとして扱わない。従って、システムは、これらの構造を優先的に回避する。例えば、“WHERE周波数=300”は“WHERE周波数/3=1000”に選択される。

・アレイのフェッチ。大きい番号の列がテーブルから選択されたとき、より早い回復のために、システムは、これらの列をアレイ内へフェッチすることを企てる。

・状態のランキング。システムがより早い回復のために、あるサーチの状態順を好むとき、サーチ状態の性質を変更して、システムが好む順のサーチ優先を割り当てるオラクルRDBMSを生じさせるフォーマット内にそれらを位置させてもよい。

・テーブル名のシーケンス。FROM節内では、“ドライビング”テーブル、交差テーブルまたは、交差テーブルがないならば、より少ない個数の記録を持つテーブル、がFROM節の端部に置かれる。FROM節内の他のテーブルは、同

様に定められる。

・状態断定シーケンス。WHERE節内では、“ドライビング”状態断定、つまり、最少の記録でリターンする最も効率のよい状態断定は、WHERE節の最初に置かれる。他の状態断定は同様に定められる。

・ブースターエンジン。WHERE節では、インデックス無しのサーチ行うために、選択された状態断定がオラクルRDBMSを生じさせるならば、第2の状態断定は、第1の実行に対してインデックスされたサーチを加え、そしてサーチされるべき記録の個数を減じる。

例えば、SQLコマンド“節の上位(パート、パートナンバー)=上位(‘DM54ALS114AJ’)から…を選択”は、各パート、パートナンバーが上位のケースに変換されなければならないので、インデックス無しのサーチを実行する。(これは単に、パート、パートナンバーでの非感知のサーチのケースである。

)そのかわり、システムは、SQLコマンド“パートWHERE上位(パート、パートナンバー)=上位(‘DM54ALS114AJ’)および(‘D%54%114%のようなパート、パートナンバー)から…を選択”を好み、すべてのアルファベットキャラクタから最初を除いたものが、サーチテキスト内のオラクルワールドカードで置き換えられる。LIKE節は、インデックスされたサーチとして実行され、そしてこれにより、より高速となり、又、非感知の場合の比較に対して扱われ、チェックされるべき記録の個数を減じる。

・別名テーブル。SELECT声明のFROM節内に記載された一つ以上のテーブルがあるときはいつでも、システムは、別名のテーブルを使用することと、それらの別名でコラム名を前に置くことを好み、より早い理解する時間を与える。

・好ましい構造。あるSQL声明は、それらの代わりのものが機能的に等価であっても、選択効率のために好まれる。例えば、一般に、NOT EXISTS構造よりNOT IN構造が選ばれ、EXISTS構造よりDISTINCT構造にが選ばれ、WHERE構造よりHAVING構造が好まれ、そして、テーブ

ルの結合よりサブの質問が好まれる。

・NOT及びORのオペレータ。オラクルRDBMSがこれらのケースでインデックス無しのテーブルスキャンを実行するので、このシステムは、NOTEQUALSのごとき、無効にされたオペレータを使用するWHERE節を好んで避ける。

SQLコマンド261は、質問モデル260と質問モデルオブジェクト9922のリストとに回答して流動的に発生され、これらのものは、ユーザー201により記述された縦方向のサーチに回答して自身で発生される。各縦方向のサーチは、一時的に全く異なり、そのため、システム200は、最初に縦方向のサーチを可行うために、SQLコマンド261を発生させ、そして次に効率的にサーチを実行できるように、最高に活用できるこれらのSQLコマンド261を与える。

当業者は、この明細書を精読した後は、システム200により発生されたSQLコマンド261に対する他のおよび別の適したものが可能であることが理解されよう。そのような他のおよび別の最適化のものは、本発明の範囲および趣旨の中にあり、そのような他のおよび別の最適化のものは別の発明や更なる努力を必要としない。

ステップ936でシステム200は、質問モデル260、質問モデルオブジェクト992、別名の記録993、下号記録994および状態記録995を試し、そして、回答して、テーブル9-3に示したものから用いたSQLコマンド261を発生させる。

テーブル9-3

“選択<結果> <テーブルおよび別名> から
どこに <テーブルの結合>
そして <状態> ”

テーブル9-3内のフォームは、質問結果251のフォームを記載するための<結果>セクション、サーチされるべきリレーショナルデータベース250内にテーブルを記載するための<テーブルおよび別名>セクション、および回復され

るべきこれらの記録により合致されるべき付随的な状態を記載するための〈状態〉を含む。〈状態〉セクションの後に、SQL“グループバイ”、“オーダーバイ”または他の声明のためのセクションのごとき、付随的なSQL声明があってもよい。

ユーザー201に示されるべき質問結果251のフォームに関する情報は、SQLコマンド261の〈結果〉セクション内に挿入される。好ましい実施例では、質問結果251のフォームに関する情報は、リレーショナルデータベース250内のテーブルから選択(SQL“選択”声明を用いて)されるべきシーケンスのコラムを含む。選択されるべきシーケンスのコラムは、縦方向のサーチを記述したとき、ユーザー201により要求されたこれらのコラムを含む。

この縦方向のサーチ例では、システム200は、クラス101 スタティクメモリを記述する結果セクションおよびサーチ可能な属性102“名前”を有する

SQLコマンド261を発生させる。このSQLコマンド261は、この結果、“選択スタティクメモリ名”のごとき声明から始める。

データを選択するためのリレーショナルデータベース内のテーブル(および別名のテーブル)に関する情報は、SQLコマンド261の〈テーブルおよび別名〉セクション内に挿入される。好ましい実施例では、テーブルおよび別名に関する情報は、リレーショナルデータベース250からサーチされるべき単一の結合テーブルへ結合されるべき、シーケンスのテーブルおよびテーブルの別名を含む。結合されるべきテーブルおよび別名の選択および順は、システム200により創造された別名記録992および結合記録993から回復される。

この縦方向のサーチ例では、システム200は、クラス スタティクメモリ およびサーチ可能な属性102“名前”を記載する結果セクションを有するSQLコマンド261を発生する。そのSQLコマンド261は、この結果、“FROMスタティクメモリ、メモリ、コンポーネント、外国メーカー、メーカー”のごとき声明で継続する。

合致すべき状態に関する情報は、SQLコマンド261の〈状態〉セクションに挿入される。好ましい実施例では、合致すべき状態に関する情報は、リレーシ

ショナルデータベース250から選択された記録により合致しなければならないシーケンスの論理オペレータ(SQL“WHERE”声明を用い、そして、多数の状態が存在するSQL“AND”声明を用いて)を含む。合致すべき状態の選択および順は、システム200により創造された状態記録994から回復される。

この縦方向のサーチの例では、システム200は、ユーザー201により要求された論理状態を記述する状態セクションを有するSQLコマンド261を、記述されたクラス101のサーチ可能な属性102を参照することにより、発生する。SQLコマンド261は、この結果、“WHERE(製造国=フランス)”のごとき声明でもって継続する。

ユーザーにより課された論理状態に加えて、〈状態〉セクションは、テーブル800の結合のためにいずれかの論理要求を課さなくてはならない。二つのタイプの結合があり、これらは、クラス101間の継承関係により要求され、又、ク

ラス101間のデータモデル関係103により要求される。

図8に関して気付くように、継承関係は、各テーブル800内のコラム810“オブジェクトID”を与えることにより、モデリングされ、派生クラス101のオブジェクトがリレーショナルデータベース250内に創造されたとき、列800は、ベースクラス101および派生クラス101の双方に、双方のテーブル800に対する前記オブジェクトに対する同じオブジェクトUIDが与えられる。この結果、継承関係は、二つのオブジェクトUIDの同一性に関連するJOIN声明と、メタモデル220内に記録された親／子の関係とにより、SQLコマンド261内でモデリングされる。

この縦方向のサーチ例では、システム200は、クラス101 スタティックメモリと、その親クラス101 メモリと、その全体の親クラス101 コンポーネントとの間の継承関係を記述する状態セクションを有する、SQLコマンド261を発生させる。SQLコマンド261は、この結果、“AND(スタティックメモリオブジェクトID=メモリオブジェクトID)AND(メモリオブジェクトID=コンポーネントオブジェクトID)”のごとき声明でもって継続する。同様に、SQLコマンド261は、“AND(メーカーのオブジェクトID=外国メ

ーカーのオブジェクトID)”のごとき声明でもって継続する。

実際には、ユーザー201がクラス101メモリのいずれかのサーチ可能な属性102との関係を作っていないので、テーブル800メモリの使用を明確に要求する必要はなく、又、前記テーブル800内のオブジェクトをマッチングさせることを要求する必要もない。システム200は、この結果、このテーブルのために継承JOINを省く。

図8に関して気付かれるように、ソースとターゲットクラス101間のデータモデル関係103は、ターゲットテーブル800内のオブジェクトを指すソーステーブル800内にコラム810を設けることによりモデリングされる、つまり、ターゲットテーブル800内の列820のオブジェクトUIDを持つ。この結果、データモデル関係103も又、オブジェクトUIDを持つポインターの同一性に関するJOIN声明と、メタモデル220内に記録されたデータモデル関係と

により、SQLコマンド261内でモデリングされる。

この縦方向のサーチ例では、システム200は、クラス101コンポーネントおよびクラス101メーカー間のデータモデル関係103を記載する状態セクションを有するAQLコマンド261を発生する。このSQLコマンド261は、この結果、“AND(コンポーネントメーカーID=メーカーオブジェクトID)”のごとき声明でもって継続する。

この結果、この縦方向のサーチ例では、SQLコマンド261は、テーブル9-4に示したごとくになる。

テーブル9-4

SM名前を選択

コンポーネントCから

スタティックメモリSM

メーカーM

外国メーカーFM

どこ(FM国=フランス)

および(M、オブジェクトID=FM、オブジェクトID)

および(C、メーカーID=M、オブジェクトID)

および(SM、オブジェクトID=C、オブジェクトID)

これらのコマンド261は、縦方向のサーチがユーザーデータベースモデル230に適用され、そしてユーザーのデータベースモデル230のユーザーリレーショナルデータベース231への翻訳に応答して、システム200により発生されることに気付く。このシステムは、200は、流動的な縦方向のサーチの翻訳が与えられ、一方では、縦方向のサーチにより使用されたモデルに適したオブジェクトとユーザーデータベースモデル230との間で、又、他方では、ユーザーのリレーショナルデータベース231により使用された関係モデルと、SQLコマンド261との間で本行くが継目なく行われる。

ステップ940でシステム200は、SQL質問をリレーショナルデータベースエンジン220に翻訳する。

ステップ950でリレーショナルデータベースエンジン220は、SQL質問261を対応するリレーショナルデータベース203に適用し、結果テーブル262を創造し、そして、結果テーブル262をユーザーインタフェイス210に翻訳する。

ステップ960でユーザーインタフェイス210は、結果テーブル262の情報をユーザー201に与える。

結果テーブル262は、関係テーブルのように、好ましくはユーザー201に与えられ、縦方向のサーチにより見付かったオブジェクトと表示のために選択されたサーチ可能な属性102を示す。

好ましい実施例では、ユーザーインタフェイス210は又、縦方向のサーチにより見付かったオブジェクトおよび、例えばスプレッドシートのフォーマットで並べて比較できるようなフォームにしてそれらのサーチ可能な属性102をユーザー201に与える。

比較フォーマットでは、ユーザーは、サーチ可能なオブジェクトの属性102を眺め、個々のオブジェクトまたは多数のオブジェクトに対するサーチ可能な属

性をエディットし、そして、選択されたオブジェクトと異なるこれらのサーチ可能な属性102を表示するがごとく、オブジェクト間で比較を行う。

フローの970ポイントで縦方向のサーチが完了する。

別の実施例。

ここで好ましい実施例が開示されたが、本発明の概念、趣旨および精神を留めた多くの変形が可能であり、これらの変形は、この明細書を精読すれば当業者には容易である。

【図1】

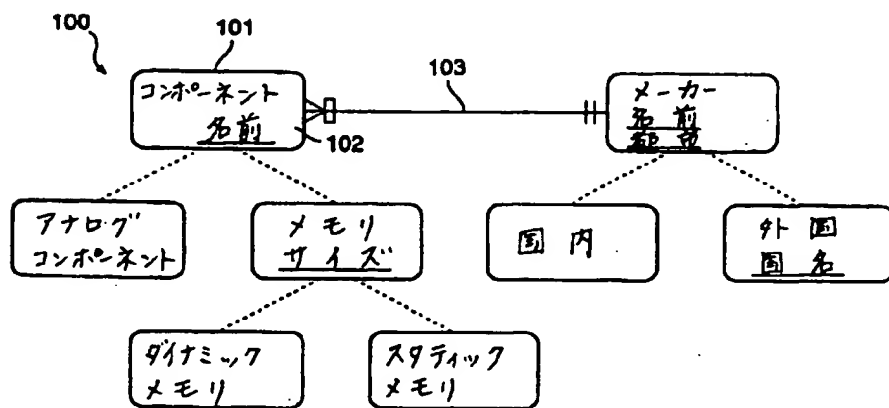


FIG. 1

【図2】

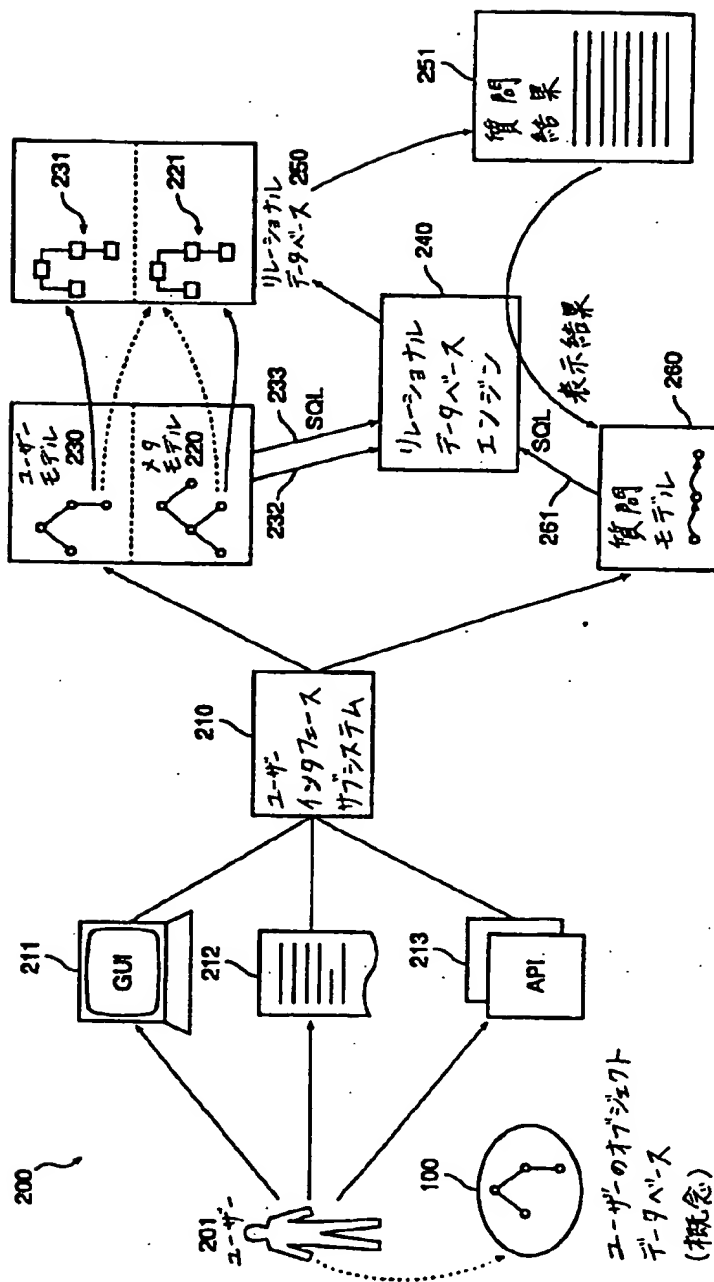
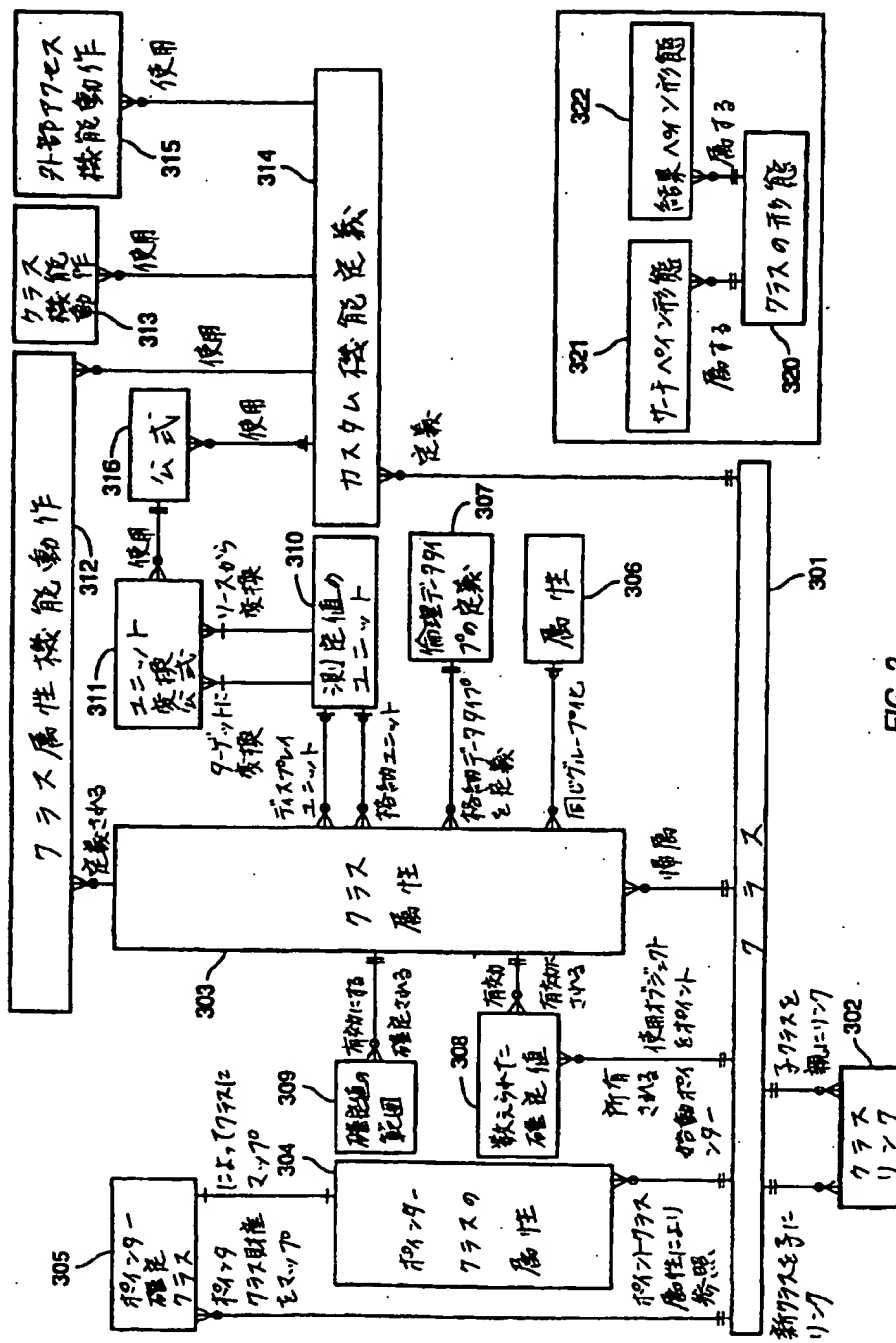


FIG. 2

FIG. 3



【図4】

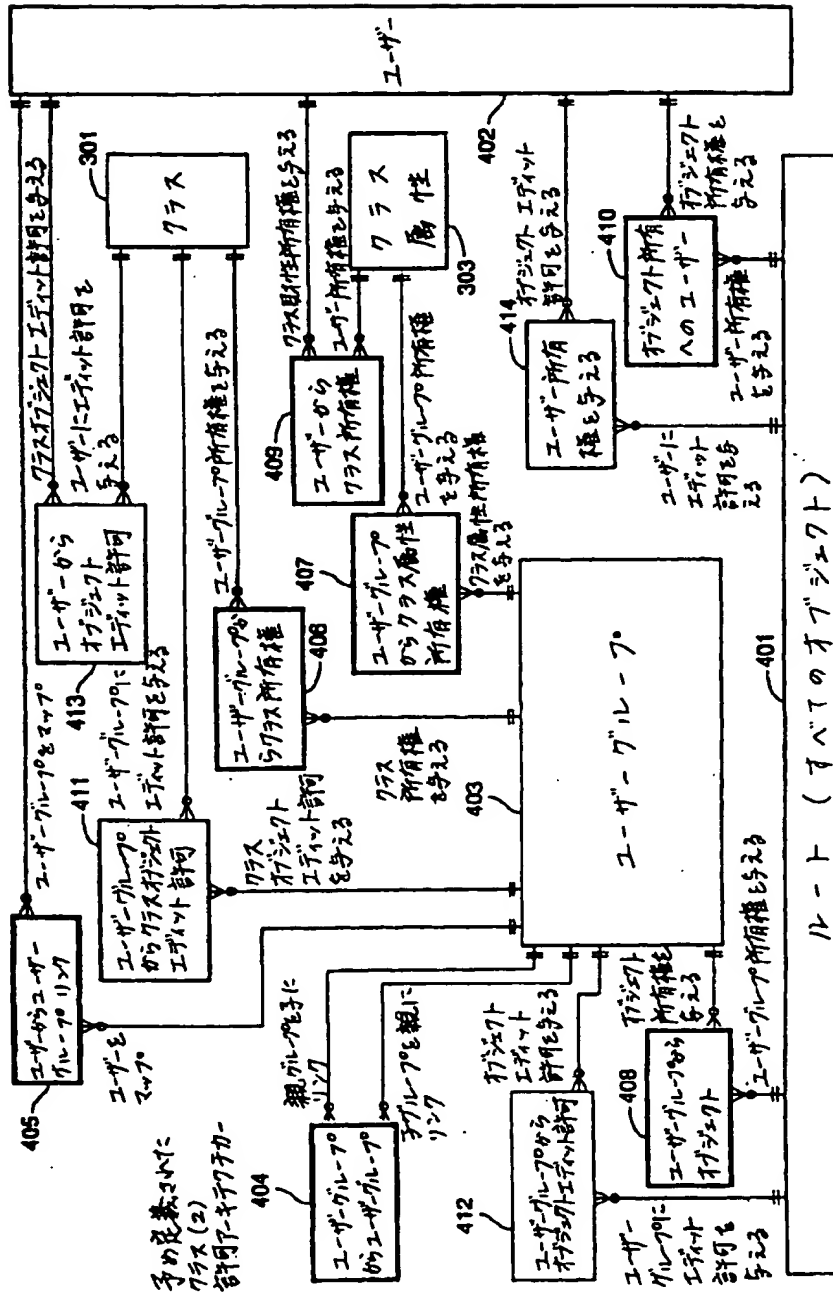
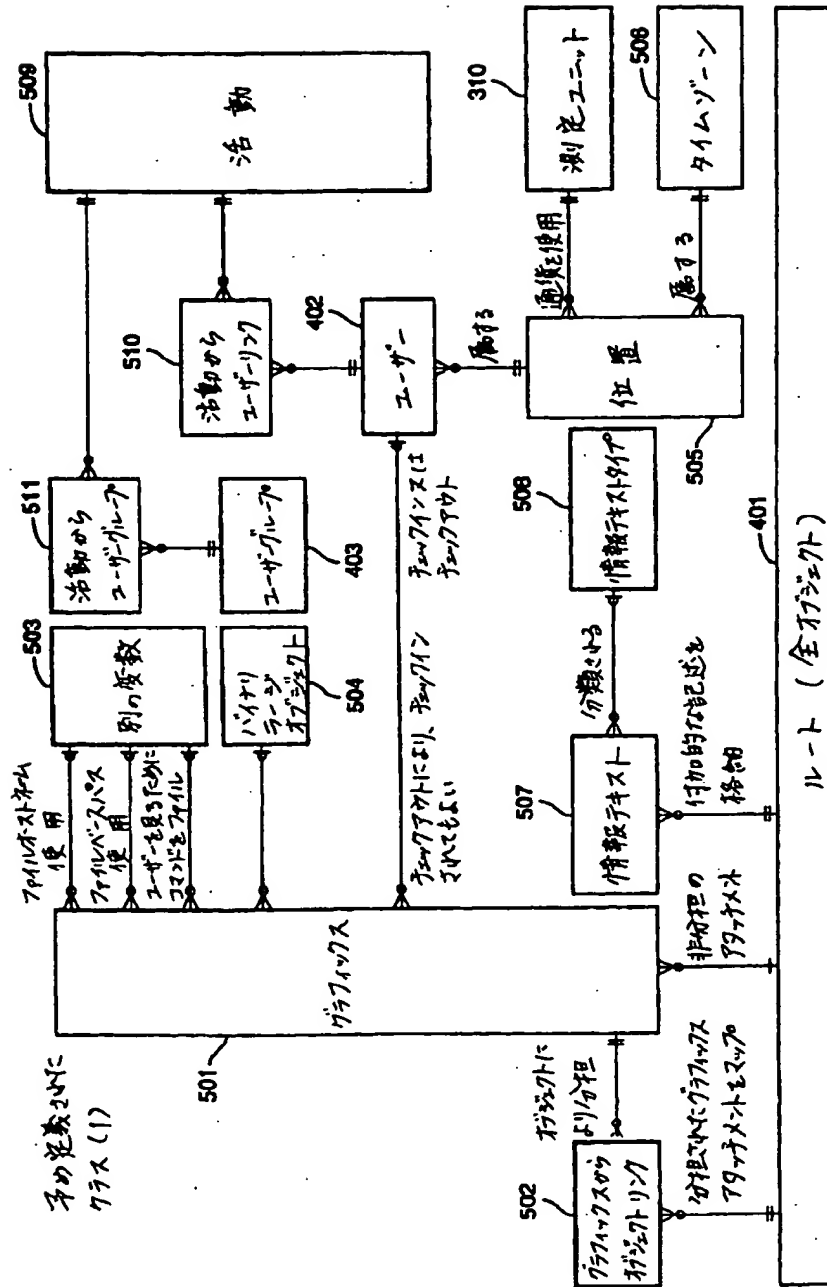


FIG. 4

FIG. 5



【図6】

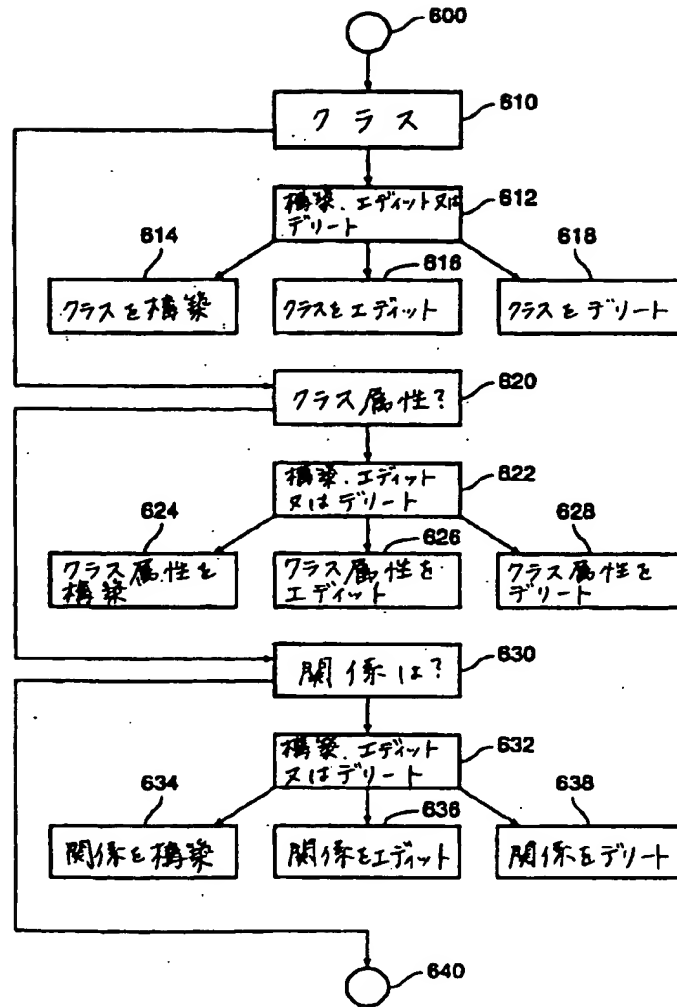


FIG. 6

【図7】

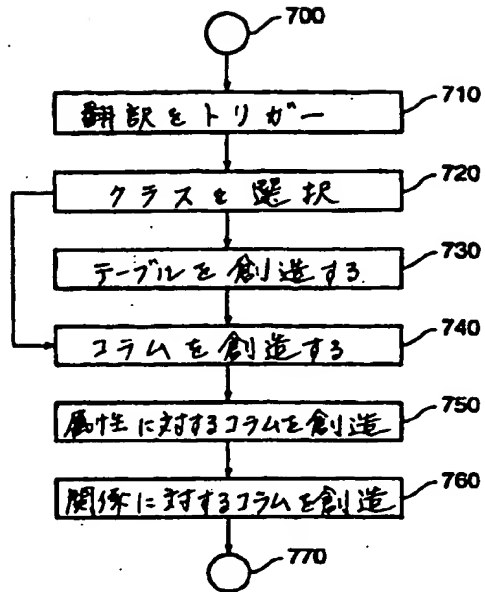


FIG. 7

【図8】

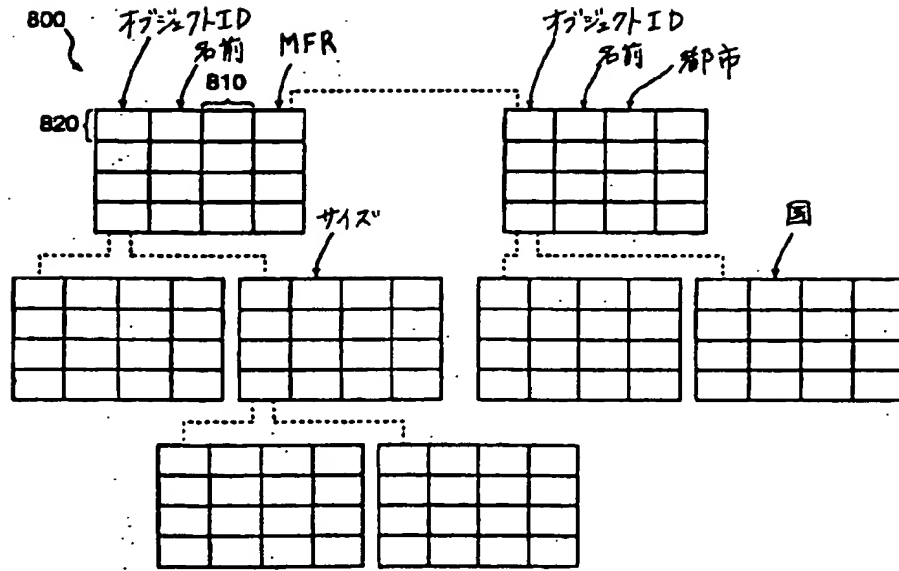


FIG. 8

【図9】

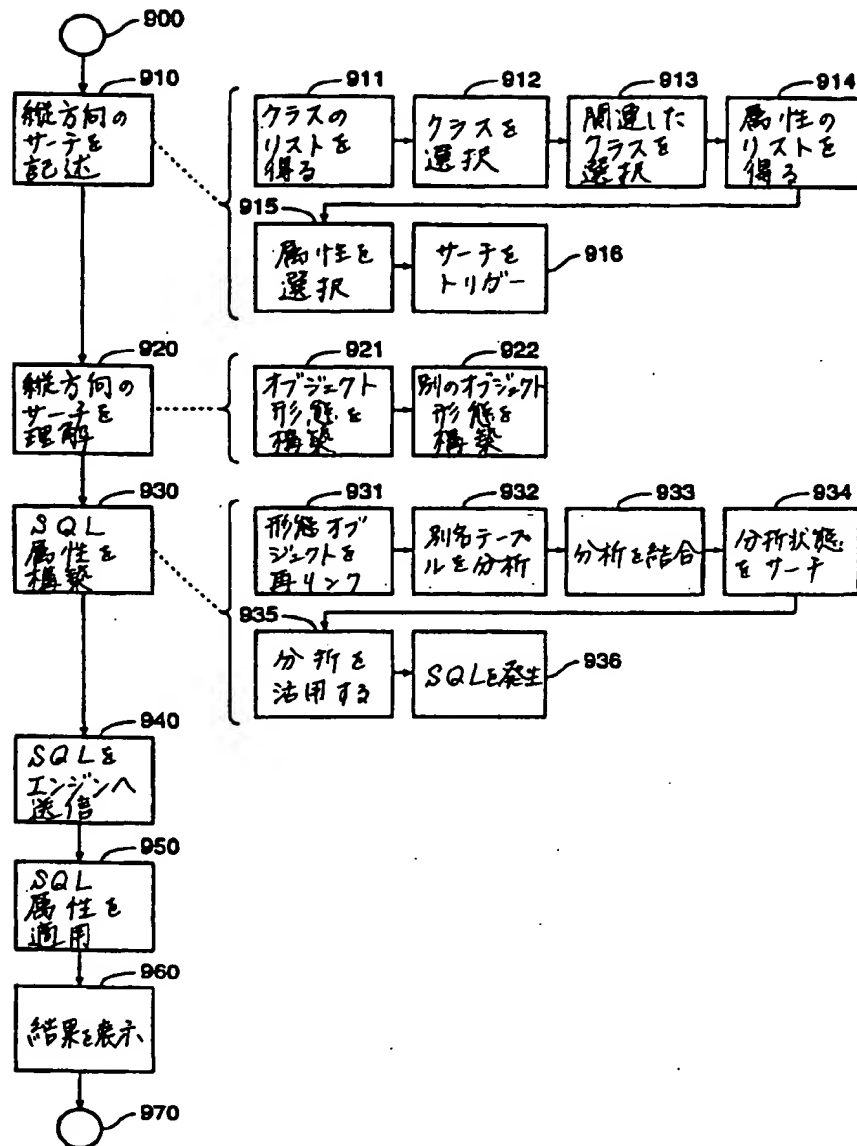


FIG. 9A

【図9】

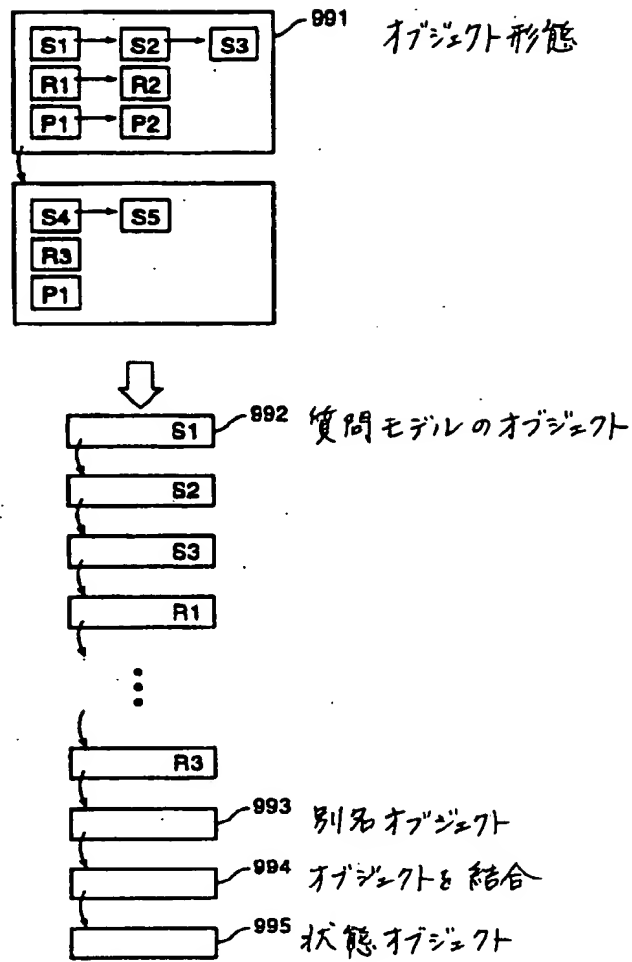


FIG. 9B

【国際調査報告】

INTERNATIONAL SEARCH REPORT

International Application No.
PCT/US 96/05678

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 G06F17/30

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC 6 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
P,X	WO,A,95 12172 (WALL DATA INC ;KROENKE DAVID M (US); OLDS CHRISTOPHER C (US); KAWA) 4 May 1995 see abstract; claims 1-36 ---	1,5,8, 10,12,16
A	WO,A,95 03586 (PERSISTENCE SOFTWARE INC) 2 February 1995 see page 3, line 7 - page 4, line 6 ---	1,5,8, 10,12,16
A	US,A,5 291 583 (BAPAT SUBODH) 1 March 1994 cited in the application see abstract see column 2, line 45 - column 3, line 62 ---	1,5,8, 10,12,16
A	US,A,5 295 256 (BAPAT SUBODH) 15 March 1994 cited in the application see column 2, line 52 - column 4, line 18 ---	1,5,8, 10,12,16
-/--		

☒ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

* Special categories of cited documents:

- * "A" documents defining the general state of the art which is not considered to be of particular relevance
- * "B" earlier document but published on or after the international filing date
- * "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- * "O" document referring to an oral disclosure, use, exhibition or other means
- * "P" document published prior to the international filing date but later than the priority date claimed

* "I" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

* "X" documents of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

* "Y" documents of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

* "A" documents member of the same patent family

Date of the actual completion of the international search

9 August 1996

Date of mailing of the international search report

09. 08. 96

Name and mailing address of the ISA
European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+ 31-70) 340 3040, Tx. 31 431 epo nl.
Fax (+ 31-70) 340 3046

Authorized officer

Katerbau, R

INTERNATIONAL SEARCH REPORT

International Application No.
PCT/US 96/05678

C(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document, with indication, where appropriate, of the relevant passage	Relevant to claim No.
A	EP,A,0 504 085 (IBM) 16 September 1992 cited in the application see abstract -----	1.5.8. 10.12.16

Form PCT 15A, 210 (continuation of second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No.
PCT/US 96/05678

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO-A-9512172	04-05-95	AU-B- 7955294 CA-A- 2175243 NO-A- 961698	22-05-95 04-05-95 26-06-96
WO-A-9503586	02-02-95	US-A- 5499371	12-03-96
US-A-5291583	01-03-94	NONE	
US-A-5295256	15-03-94	NONE	
EP-A-0504085	16-09-92	US-A- 5212787	18-05-93

Form PCT/ISA/210 (patent family members) (July 1992)

フロントページの続き

(81)指定国 EP(AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), CA, FI, JP, KR

(72)発明者 ビランガー, ケン
アメリカ合衆国03051ニュー・ハンプシャー州ハドソン、シェイファー・サークル24番

(72)発明者 ブラサド, ナゲンドラ
インド、バンガロール、パドマナバナゲル、サーティーンズ・メイン472番

(72)発明者 マクギニス, ブライアン
アメリカ合衆国94526カリフォルニア州ダンビル、クラリタ・プレイス11131番

(72)発明者 マクウィリアムズ, フロイド
アメリカ合衆国94031カリフォルニア州パロ・アルト、ホーソーン・アベニュー・ナンバー248、275番

(72)発明者 ツァン, ヨン
アメリカ合衆国94086カリフォルニア州サニーベイル、エスカロン・アベニュー・ナンバー901、1055番

(72)発明者 コウシク, ラビ
アメリカ合衆国94040カリフォルニア州マウンテン・ビュー、エスクエラ・アベニュー316番 アpartment・ナンバー91

【要約の続き】

らのクラスの各々は1組みのサーチ可能な属性を含み、これらの関係の各々は、(ベースクラスと派生クラスとの間の)継承関係か、(一つから一つ、一つから多くまたは多くから多くの関係)データモデル関係を含んでもよい。ユーザーのオブジェクトデータベースのデータモデルは、メタモデル内の実際のオブジェクトによりモデリングされ、そして、ユーザーのオブジェクトデータベースのエディットまたは扱いは、メタモデル内のオブジェクトを創造し、変形し、あるいは削除することによりモデリングされる。このメタモデルは又、それがユーザーのオブジェクトデータベースをモデリングしたように自身をモデリングし、又、ユーザーのオブジェクトデータベースと同じ方法で扱われる。